Caracterización y predicción automática de dificultad en colecciones de Búsqueda de Respuestas en base a Modelos Neuronales de Lenguaje

Lara Olmos Camarena

Máster en Tecnologías del Lenguaje

Universidad Nacional de Educación a Distancia (UNED)

Junio 2021

Dirigido por:

Álvaro Rodrigo Yuste

Agradecimientos

Gracias a Álvaro Rodrigo por la motivación de realizar este trabajo y con ello dar un paso más allá donde las preguntas y respuestas no acaban sobre los Modelos Neuronales de Lenguaje pre-entrenados.

Gracias a mi familia y amigos, una vez más, por todo el cariño y apoyo.

En especial, a todo un equipo de *cracks* con los fue un placer coincidir laboralmente al comienzo del estudio del máster. Nacho, Ulises, Rafa, César, Manuel, Alejandro, Pablo, Pedro y Berta, y continúa la lista... gracias por la oportunidad de avistar el área de Procesamiento de Lenguaje Natural con *geniecillos*, semilla de curiosidad para este trabajo, y los que vendrán...

Palabras, tan próximo mi saber, que me acompañas
tan unido mi esencia, tan libre
con mis rinsensas, chelo de vida al viento,
los sueños, hechos nuestros deseos.
Transcurre en nuestras venas,
circula sin límites de velocidad,
posee todos los que vendrán por las puertas,
caminos de todos los pequeños de algas,
¿para cuándo son los de las estrellas?

(Olmos Camarena, Lara. *Caminos de estrellas*. Marzo-Abril 2019. Generado con *BiLSTMs* entrenadas sobre el corpus de *LMDL*)

Resumen

El área de Búsqueda de Respuestas evoluciona gracias al uso de colecciones que permiten la evaluación del rendimiento de los sistemas ante la necesidad de conocimiento general. Los últimos sistemas de Deep Learning, basados en modelos pre-entrenados como *BERT*, *RoBERTa* y *T5*, han mejorado en gran medida los resultados de planteamientos anteriores. Sin embargo, el análisis de errores cometidos por estos sistemas es escaso y no permite conocer en qué aspectos se puede mejorar o qué tipo de preguntas plantean mayor dificultad.

Para abordar este problema, en este trabajo se ha realizado una caracterización automática de las colecciones más empleadas como *SQuAD*, *NewsQA* y *RACE*, y un estudio asociando los fallos y aciertos cometidos por varios modelos sobre estas colecciones. Además, se propone una metodología para la anotación automática de la complejidad de las colecciones de preguntas en base a las dificultades que suponen para varios sistemas. Finalmente, se evalúan varios modelos predictivos basados en Aprendizaje Automático para estudiar la capacidad de predecir la anotación propuesta en este trabajo. De este modo, se pretende avanzar en los estudios relativos a cómo mejorar los resultados en la Búsqueda de Respuestas por parte de los sistemas actuales.

Palabras clave

Búsqueda de Respuestas, Modelos Neuronales de Lenguaje, Procesamiento de Lenguaje Natural, Aprendizaje Automático Profundo, *Transfer Learning*

Abstract

Question answering evolution is due to the explosion of challenging datasets requiring world knowledge to answer. Recently, pre-trained neural network language models such as *BERT*, *RoBERTa* and *T5* have greatly improved on the results of previous approaches. However, error analysis of this models is scarce and and does not allow to know in which aspects can be improved or what type of questions pose the greatest difficulty.

To address this problem, in this work is proposed the automatic linguistic characterization of several datasets used for fine-tunning this models such as *SQuAD*, *NewsQA* and *RACE*, and a study associating the mistakes and successes made by various models on these collections. In addition, a methodology for automatic annotation of the complexity of question collections is proposed based on the difficulties the pose to various systems. Finally, several predictive models based on Machine Learning are evaluated to study the ability to predict the annotation proposed in this work. In this way, it is intended to advance in the studies on how to improve the results in Question Answering by the current systems.

Keywords

Question Answering, Neural Networks Language Models, Natural Language Processing, Deep Learning, Transfer Learning

Índice general

1 Introducción **1**

1.1 Objetivos 6

1.2 Estructura del documento 7

2 Estudio del estado del arte **10**

2.1 Búsqueda de Respuestas 11

2.1.1 Contextualización histórica 14

2.1.2 Comprensión Lectora 16

2.2 Colecciones de Pregunta-Respuesta 18

2.2.1 MCTest 19

2.2.2 CNN Daily Mail 20

2.2.3 SQuAD 23

2.2.3.1 Modificación ruidosa de SQuAD 24

2.2.4 RACE 27

2.2.5 NewsQA 29

2.2.6 CoQA 32

2.2.7 QuAIL 36

2.3 Modelos Neuronales de Lenguaje en Búsqueda de Respuestas 39

2.3.1 DrQA 41

2.3.2 BiDAF 42

2.3.3 Modelos Neuronales de Lenguaje Pre-Entrenados . . . 43
2.3.3.1 GPT . . . 48
2.3.3.2 BERT . . . 52
2.3.3.3 RoBERTa . . . 54
2.3.3.4 XLNet . . . 55
2.3.3.5 T5 . . . 57
2.4 Análisis de dificultad preliminar . . . 59
2.4.1 Análisis y comparativa de las colecciones . . . 61
2.4.2 Dificultades de los Modelos Neuronales de Lenguaje . 63
2.5 Métodos de Procesamiento de Lenguaje Natural . . . 66
2.5.1 Etiquetadores léxicos y análisis sintáctico . . . 67
2.5.2 Reconocimiento de entidades . . . 70
2.5.3 Obtención del foco de la pregunta . . . 74
3 Caracterización de colecciones de Pregunta-Respuesta 77
3.1 Estudio de fenómenos lingüísticos . . . 78
3.2 Análisis por caracterización lingüística . . . 80
3.2.1 Análisis de las preguntas . . . 82
3.2.1.1 Análisis sintáctico de las preguntas . . . 83
3.2.1.2 Análisis del foco de las preguntas . . . 86
3.2.1.3 Análisis de coincidencias entre preguntas y contextos . . . 87
3.2.2 Análisis de las respuestas . . . 89
3.2.2.1 Análisis sintáctico de las respuestas . . . 90
3.2.2.2 Análisis de las entidades de las respuestas . . 90

3.2.3 Análisis por asociación entre foco de la pregunta y tipo de respuesta . . . 93
3.3 Análisis por resultados obtenidos con los modelos . . . 95
3.3.1 Análisis de errores por longitudes . . . 98
3.3.2 Análisis de errores por tipos de preguntas . . . 101
3.3.3 Análisis de errores por tipos de respuestas . . . 105
3.3.4 Análisis de errores en clases relativas al foco de la pregunta y entidad de la respuesta . . . 107
3.3.5 Análisis de la confianza de respuesta . . . 108
3.4 Análisis de dificultad por caracterización de las colecciones . . . 110

4 Metodología para la anotación automática de dificultad 115

4.1 Criterios de comparación entre respuestas . . . 116
4.2 Anotación por errores binaria (binario ingenuo) . . . 117
4.3 Anotación por dificultad binaria por mayoría . . . 118
4.4 Anotación por dificultad multiclase . . . 118

5 Resultados de la anotación automática de dificultad 120

5.1 Anotación binaria ingenua . . . 122
5.2 Anotación binaria por mayoría . . . 124
5.3 Anotación por dificultad multiclase . . . 125

6 Predicción de dificultad 127

6.1 Variables de dificultad por caracterización lingüística . . . 129
6.1.1 Importancia de las variables por resultados obtenidos con los modelos . . . 133

6.2 Modelo de predicción de dificultad 137
6.3 Evaluación de dificultad con modelos con especialización sobre *SQuAD* 139
6.3.1 Modelo de predicción de dificultad binario ingenuo . . 139
6.3.2 Modelo de predicción de dificultad binario por mayoría 141
6.3.3 Modelo de predicción de dificultad multiclase 143
6.4 Evaluación de dificultad con modelos con especialización sobre *NewsQA* 145
6.4.1 Modelo de predicción binario ingenuo 146
6.4.2 Modelo de predicción de dificultad binario por mayoría 147
6.4.3 Modelo de predicción de dificultad multiclase 149

7 Conclusiones y trabajos futuros **153**

Bibliografía **163**

A Anexo 1. Fases y desarrollo del trabajo **165**
A.1 Fases del trabajo 165
A.2 Desarrollo 167
A.2.1 Análisis sintáctico 167
A.2.2 Reconocimiento de entidades 169
A.2.3 Obtención de foco de la pregunta 171
A.2.4 Desarrollo para ejecución de modelos sobre colecciones de Pregunta-Respuesta 172
A.2.5 Desarrollo para asociación foco de la pregunta y entidad de la respuesta 173

A.2.6 Desarrollo para diferencia entre respuestas y detección de errores 175

A.2.7 Desarrollo para modelos de predicción 177

B Anexo 2. Detalle de caracterización de colecciones de Pregunta-Respuesta 179

B.1 Reconocimiento de entidades sobre colecciones de Pregunta-Respuesta 180

B.2 Ejemplos de fenómenos lingüísticos 183

B.2.1 Ejemplo de fenómeno de contextualización errónea por escasez de términos comunes 183

B.2.2 Ejemplo de confusión por términos comunes y/o coreferencia 184

B.2.3 Ejemplo de dificultad de selección del comienzo o final de la respuesta 184

B.2.4 Ejemplo de estructura sintáctica complicada 185

B.2.5 Ejemplo de procesamiento de múltiples oraciones y razonamiento 186

B.2.6 Ejemplo de necesidad de conocimiento externo, inferencia y entendimiento 187

B.2.7 Ejemplo de tratado distinto o erróneo del lenguaje . . 188

B.2.8 Ejemplo de dificultad de la respuesta por tipo o entidad 188

B.2.9 Ejemplo de dificultad de selección de la respuesta correcta entre varias 189

B.2.10 Ejemplo de fenómeno de complejidad y ambigüedades que hacen dudar también a una persona 190
B.3 Errores cometidos por los modelos en base a la caracterización lingüística 192
B.3.1 Errores cometidos por clase de foco de la pregunta y tipo de entidad de respuesta 193

Índice de figuras

2.1 Ejemplo de *AddOneSent*, en azul oración adversa añadida, en rojo fallos. . . 26
2.2 Ejemplo de *RACE* evidencia de opinión del autor del texto, pregunta y múltiples respuestas (Lai et al., 2017) . . . 28
2.3 Evaluación con *RACE* para *GPT*, *BERT*, *RoBERTa* y *XLNet* Yang et al. (2019) . . . 29
2.4 Ejemplo de *CoQA*, en color coreferencias de la conversación (Reddy et al., 2019) . . . 33
2.5 Distribución de trigramas de inicio de *SQuAD 2.0* y *CoQA* (Reddy et al., 2019) . . . 35
2.6 Origen de los modelos neuronales de lenguaje: *feed forward newtork* (Bengio et al., 2003) . . . 44
2.7 Arquitectura de *Transformer* (Vaswani et al., 2017) . . . 46
2.8 Modelos de la rama de estudio de *Bertology* (Wang and Zhang, 2019-2021) 47
2.9 Arquitectura de *GPT-1* y entradas para múltiples tareas de Procesamiento de Lenguaje (Radford et al., 2018) . . . 49
2.10 Preguntas y respuestas generadas con *GPT-2* con mayor confianza (Radford et al., 2019) . . . 51
2.11 *Embeddings* empleados en *BERT* (Devlin et al., 2018) . . . 53
2.12 Procesos de pre-entrenamiento y refinamiento de *BERT* (Devlin et al., 2018) 53
2.13 Comparación entre *RoBERTa*, *BERT*, *XLNet* y su evaluación sobre *SQuAD* (Liu et al., 2019) . . . 54
2.14 Ejemplo de factorización de *BERT* y de *XLNet* (Yang et al., 2019) . . . 56
2.15 Comparación entre *RoBERTa*, *BERT*, *XLNet* y su evaluación *GLUE* (Yang et al., 2019) . . . 56
2.16 Evaluación con *SQuAD* para *BERT*, *XLNet* y *RoBERTa* (Yang et al., 2019) 56
2.17 Planteamiento de *Text-to-Text Transfer Transformer*, T5 (Raffel et al., 2019) 57

3.1 Número de entidades anotadas en *SQuAD v2.0 train* en las preguntas, contextos y respuestas . . . 88
3.2 Relación entre la longitud de la pregunta, respuesta y texto para *SQuAD v2.0 train* . . . 99
3.3 Relación entre la longitud de la respuesta para *SQuAD v2.0 train y dev* . . 100

3.4 Relación entre la longitud de la pregunta, respuesta y texto para *SQuAD v2.0 dev* . . . 100
3.5 Relación entre el tipo de partícula interrogativa y errores cometidos en *SQuAD v2.0 train y dev* . . . 102
3.6 Relación entre el tipo de trigrama morfosintáctico y errores cometidos en *SQuAD v2.0 train y dev* . . . 103
3.7 Relación entre el tipo de foco de la pregunta y errores cometidos en *SQuAD v2.0 train y dev* . . . 104
3.8 Relación entre el tipo de respuesta sintáctica y errores cometidos en *SQuAD v2.0 train y dev* . . . 106
3.9 Relación entre el tipo de respuesta por entidad reconocida y errores cometidos en *SQuAD v2.0 train y dev* . . . 106
3.10 Relación entre el foco de la pregunta y tipo de respuesta por entidad reconocida en *SQuAD v2.0 train y dev* . . . 108
3.11 Distribución de confianza de respuesta (KDE) para *SQuAD train* (A) y *dev* (C) para *BERT-base*, *BERT-large*, *RoBERTa-base* y *RoBERTa-large*. Análogo para partición con errores de *SQuAD train* (B) y *dev* (D). . . . 109

6.1 Correlación de las variables por caracterización lingüística para la colección *SQuAD train* . . . 134
6.2 Importancia de las variables por caracterización lingüística para la colección *SQuAD train* . . . 134
6.3 Importancia de las variables por caracterización lingüística para la colección *SQuAD train* . . . 136
6.4 Correlación de las variables por caracterización lingüística para las preguntas contestables de la colección *SQuAD train* . . . 136

B.1 Tipo de entidades reconocidas en las respuestas de las colecciones *SQuAD*, *NewsQA* y *RACE* . . . 180

Índice de tablas

2.1 Análisis de consideraciones lingüísticas de pregunta y respuesta en CNN Daily Mail datasets (Chen et al., 2016). 22

2.2 Tipos de preguntas no contestables y proporción de preguntas de una muestra de 100 ejemplos (Rajpurkar et al., 2018) 25

2.3 Tipos de razonamientos y proporción de preguntas de una muestra de 1000 ejemplos de *SQuAD* y *NewsQA* (Trischler et al., 2017) 30

2.4 Tipos de respuestas de *SQuAD* (Rajpurkar et al., 2016) 31

2.5 Tipos de respuestas de *NewsQA* (Trischler et al., 2017) 31

2.6 Dominios de *CoQA* (Reddy et al., 2019) 34

2.7 Fenómenos lingüísticos de una muestra de 150 ejemplos de *CoQA* (Reddy et al., 2019) 36

2.8 Tipos de pregunta y exactitud obtenida en la evaluación del rendimiento humano en *QuAIL* (Rogers et al., 2020) 38

2.9 Evaluación de *Text-to-Text Transfer Transformer*, T5, en *GLUE* y *SQuAD* (Raffel et al., 2019). Eficiencias con modelos basados en BERT (a) ALBERT, (b) StructBERT, (c) XLNet. 59

2.10 Comparativa cualitativa de las colecciones de Pregunta-Respuesta 62

2.11 Comparativa cuantitativa de las colecciones de Pregunta-Respuesta 63

2.12 Información estadística de razonamientos y aspectos lingüísticos en las colecciones *RACE*, *CNN Daily Mail* (Chen et al., 2016) en comparación con *SQuAD* (Lai et al., 2017) 63

2.13 Exactitud obtenida con modelos basados en *Transformer* sobre *SQuAD v1.1* de test (dev), Aspillaga et al. (2020). 65

2.14 Etiquetas morfosintácticas de *Penn Treebank* (P. Marcus et al.) sobre ejemplos de palabras de comienzo de preguntas 69

2.15 Eficiencia obtenida para reconocimiento de entidades con *Stanza*, *Flair* y *SpaCy* (Qi et al., 2020) 71

2.16 Ejemplos de anotación NER con *SpaCy* sobre documentos de *SQuAD* . . . 72

2.17 Tipos de entidades disponibles en *CoreNLP*, *spaCy* y *Stanza* 73

2.18 Resumen de la taxonomía de preguntas Moldovan (Moldovan et al., 2000), (Martínez-Barco et al., 2014) 75

2.19 Ejemplos de pregunta, focos y tipo de entidad de la respuesta sobre ejemplos de *SQuAD* . . . 75
2.20 Ejemplos de asociación entre foco de la pregunta y tipo de entidad de la respuesta (NER) para preguntas de hechos concretos . . . 76

3.1 Análisis de longitud media de las preguntas, respuestas (*no nulas) y contextos de las colecciones de Pregunta-Respuesta *SQuAD* (*train*, *dev*), *NewsQA* (contextos de *CNN Daily Mail*) y *RACE* para *train*, *test* y *dev* . . . 81
3.2 Comparativa por palabra de comienzo de las preguntas de las colecciones de Pregunta-Respuesta *SQuAD*, *NewsQA* y *RACE* . . . 83
3.3 Comparativa por estructura sintáctica de toda la pregunta de las colecciones de Pregunta-Respuesta *SQuAD*, *NewsQA* y *RACE* . . . 84
3.4 Comparativa de número de preguntas por trigrama de etiquetas morfosintácticas al comienzo de las preguntas de las colecciones de Pregunta-Respuesta *SQuAD*, *NewsQA* y *RACE* . . . 85
3.5 Comparativa por foco de la pregunta de las colecciones de Pregunta-Respuesta *SQuAD*, *NewsQA* y *RACE* . . . 87
3.6 Top 10 de entidades reconocidas de las preguntas y en los textos de las colecciones de Pregunta-Respuesta en entrenamiento . . . 89
3.7 Comparativa por formulación de las respuestas las colecciones de Pregunta-Respuesta *SQuAD* y *NewsQA*, *incluyendo respuestas a preguntas *null* . . 90
3.8 Comparativa por entidades reconocidas en las respuestas de Pregunta-Respuesta *SQuAD* y *NewsQA* . . . 91
3.9 Comparativa por número medio de entidades en los textos de Pregunta-Respuesta del mismo tipo a la entidad esperada como respuesta para *SQuAD* y *NewsQA* . . . 92
3.10 Comparativa por asociación del foco de la pregunta y tipo de respuesta por sus entidades reconocidas de colecciones de Pregunta-Respuesta *SQuAD* y *NewsQA*, *incluyendo respuestas a preguntas *null* o sin entidades reconocidas 94
3.11 Modelos de HuggingFace para Pregunta-Respuesta con fine-tunning con *SQuAD* . . . 96
3.12 Comparativa por errores cometidos sobre preguntas no *null* por los modelos con especialización en la colección de Pregunta-Respuesta *SQuAD* . . . 97

5.1 Modelos de HuggingFace para Pregunta-Respuesta con fine-tunning en *NewsQA* . . . 122
5.2 Anotación binaria ingenua con Modelos Neuronales de Lenguaje con *fine-tunning* sobre *SQuAD* preguntas *no null* . . . 123
5.3 Anotación binaria ingenua con Modelos Neuronales de Lenguaje con *fine-tunning* sobre *NewsQA* sobre preguntas *no null* . . . 123

5.4 Anotación binaria por mayoría con Modelos Neuronales de Lenguaje con *fine-tunning* sobre *SQuAD* preguntas *no null* . . . 124
5.5 Anotación binaria por mayoría con Modelos Neuronales de Lenguaje con *fine-tunning* sobre *NewsQA* sobre preguntas *no null* . . . 125
5.6 Anotación multiclase con Modelos Neuronales de Lenguaje con *fine-tunning* sobre *SQuAD* preguntas *no null* . . . 126
5.7 Anotación multiclase con Modelos Neuronales de Lenguaje con *fine-tunning* sobre *NewsQA* sobre preguntas *no null* . . . 126

6.1 Variables finales de los modelos de predicción de dificultad . . . 132
6.2 Evaluación del modelo de dificultad binario ingenuo sobre *SQuAD dev* entrenado con *SQuAD train* . . . 140
6.3 Evaluación del modelo de dificultad binario ingenuo sobre *NewsQA test* entrenado con *SQuAD train* . . . 140
6.4 Exactitud (*accuracy*) del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo sobre SQuAD dev 2.0 . . . 141
6.5 Evaluación del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 para entrenamiento con *SQuAD* train . . . 142
6.6 Evaluación del modelo de dificultad binario sobre NewsQA test entrenado con SQuAD train 2.0 . . . 142
6.7 Exactitud (*accuracy*) del modelo de dificultad binario por mayoría sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 . . . 143
6.8 Evaluación del modelo de dificultad multiclase sobre SQuAD dev 2.0 . . . 143
6.9 F1-score del modelo de dificultad multiclase sobre SQuAD dev . . . 144
6.10 Evaluación del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0 . . . 144
6.11 F1-score del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0 . . . 144
6.12 Exactitud (*accuracy*) del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo de dificultad multiclase sobre SQuAD dev . . . 145
6.13 Evaluación del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con NewsQA train . . . 146
6.14 Evaluación del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con SQuAD dev 2.0 . . . 146
6.15 Exactitud (*accuracy*) del modelo de dificultad binario ingenuo sobre SQuAD test entrenado con NewsQA train. Evaluación del modelo sobre NewsQA test 147

6.16 Evaluación del modelo de dificultad binario por mayoría sobre NewsQA test entrenado con *NewsQA train* . . . 148
6.17 Evaluación del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 test entrenado con NewsQA train . . . 148
6.18 Exactitud (*accuracy*) del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 test entrenado con NewsQA train. Evaluación del modelo de dificultad binario por mayoría sobre NewsQA test . . . 149
6.19 Evaluación del modelo de dificultad multiclase sobre NewsQA test entrenado con *NewsQA train* . . . 149
6.20 F1-score del modelo de dificultad multiclase sobre NewsQA test . . . 150
6.21 Evaluación del modelo de dificultad multiclase sobre SQuAD 2.0 dev entrenado con NewsQA train . . . 150
6.22 F1-score del modelo de dificultad multiclase sobre SQuAD dev 2.0 entrenado con NewsQA train . . . 150
6.23 Exactitud (*accuracy*) del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo de dificultad multiclase sobre SQuAD dev 2.0 . . . 151

B.1 Número de entidades reconocidas con *CoreNLP* en las preguntas de las colecciones de Pregunta-Respuesta *SQuAD 2.0*, *NewsQA* y *RACE* . . . 181
B.2 Número por tipo de entidades reconocidas con *CoreNLP* en las respuestas de las colecciones de Pregunta-Respuesta *SQuAD 2.0*, *NewsQA* y *RACE* . . 182
B.3 Errores y aciertos en preguntas no *null* de modelos basados en *BERT* (*phiyodr/bert-base-finetuned-squad-v2*, *phiyodr/bert-large-finetuned-squad-v2*), *RoBERTa* (*deepset/roberta-base-squad2*, *phiyodr/roberta-large-finetuned-squad2*) y *T5* (*valhalla/t5-base-squad*) para *SQuAD* . . . 192
B.4 Comparativa de errores en base a clase de foco de la pregunta y tipo de respuesta sobre preguntas con respuesta de *SQuAD 2.0 dev* . . . 193
B.5 Comparativa de errores en base a clase de foco de la pregunta y tipo de respuesta sobre preguntas con respuesta de *SQuAD 2.0 train* . . . 194

Capítulo 1

Introducción

Actualmente, gran cantidad de información está presente en formato digital y es accesible gracias a los motores de búsqueda, libros, medios de comunicación, enciclopedias, artículos, publicaciones, foros y redes sociales disponibles online. Los usuarios acceden a la información mediante la indagación, localización y comprensión de documentos o fuentes de información textuales cada vez más masivas y heterogéneas, incluso multimedia con soporte a transcripción y traducción para lidiar con múltiples formatos e idiomas.

El área de estudio de la Búsqueda de Respuestas o sistemas de Pregunta-Respuesta surge con la necesidad de procesar esta gran cantidad de información digital y dar respuesta rápida, precisa y escueta a usuarios con distintos niveles de experiencia y conocimiento que formulan preguntas arbitrarias en un lenguaje cotidiano (Vicedo, 2004).

Para ello, se requiere el análisis de la pregunta y la recuperación de los documentos o fuentes de información (selección de pasajes candidatos a contener la respuesta, su filtrado y orden por relevancia), la extracción de datos y contextualización para formar la respuesta por su tipo (un pasaje, una definición, hecho, persona, lugar, etc.) y forma (una respuesta, sin respuesta, multirespuesta) que de forma natural un ser humano realiza en la comprensión del lenguaje natural (Moldovan et al., 2000).

Estos sistemas incluyen técnicas de Procesamiento de Lenguaje Natural que permiten anotar el lenguaje para su procesado automático en distintas fases consecutivas (*pipelines*). Desde el análisis tanto a nivel de palabra (tras la separación de oraciones y palabras, *tokenización*) con el análisis morfosintáctico se permite contemplar la relevancia y el cambio de significado de cada una por la estructura de la oración y así componer más información de entrada a sistemas de representación de lenguaje (Jurafsky and Martin, 2009-2021). Posteriormente, reconocer entidades propias (personas, organizaciones, lugares, etc.), referencias temporales u otras entidades (fechas, cantidades monetarias, unidades de sistemas métricos, porcentajes, *urls*, etc.) y extraer relaciones (analogías, jerarquías) y coreferencias, permiten contextualizar, representar conocimiento y semántica presente en un texto (Camacho-Collados and Pilehvar, 2018). También, gracias a los métodos estadísticos y modelos probabilísticos sus técnicas permiten el análisis de estructuras frecuentes a nivel de *n-gramas* (palabras que aparecen de forma consecutiva), el análisis de frecuencias y análisis de datos para su caracte-

rización, extracción de temáticas principales y clasificación a partir de la representación de textos con modelos de Aprendizaje Automático y Deep Learning (Collobert et al., 2011).

Por tanto, relacionada con el área de Acceso y Extracción de Información y Procesamiento de Lenguaje Natural, la Búsqueda de Respuestas ha evolucionado como ámbito independiente con tareas asociadas como la búsqueda de respuestas frecuentes, comprensión de textos o Comprensión Lectora (con exámenes y tests) y Búsqueda de Respuestas en dominios específicos (noticias, artículos, medicina) hasta de ámbito y conocimiento general, (Clark and Etzioni, 2016). Actualmente, sus arquitecturas, métodos y técnicas se emplean en buscadores web, repositorios documentales y en sistemas de indexación de contenidos hasta los actuales sistemas de diálogo, en ámbito comercial denominados *chatbots* o asistentes virtuales.

Estos sistemas de Búsqueda de Respuestas se enfrentan a formulaciones de nuevas preguntas en lenguaje natural con las que no se ha entrenado lidiando con múltiples fenómenos lingüísticos (desde errores ortográficos, paráfrasis hasta la similitud semántica de textos) junto con la necesidad de procesado de múltiples fuentes de información para su enriquecimiento y representación de significado (Camacho-Collados and Pilehvar, 2018). Estos retos se superan tras la formación de un corpus para obtención de palabras clave, extracción de relaciones, taxonomías, ontologías, redes semánticas o grafos de conocimiento combinadas con el uso de técnicas de indexación, clasificación o agrupación por temáticas (Ferrucci et al., 2010) y recientemente

representaciones abstractas con *Deep Learning* para extracción, selección y generación de respuestas (Manning, 2020). Todo ello en arquitecturas que permitan un tiempo de respuesta adecuado en tiempo real, lidiando con distintos idiomas, perfiles de usuario y de forma interactiva para mejorar su resultado (Martínez-Barco et al., 2014).

Por tanto, el área de estudio de la Búsqueda de Respuestas tiene como objetivo no solo localizar la fuente de origen de la respuesta sino la evidencia de forma precisa dada una pregunta, idealmente mediante su contextualización, selección, comprensión o generación a partir de un texto tras el análisis de la pregunta (Clark and Etzioni, 2016). Múltiples de estas tareas han sido superadas con modelos del área de Deep Learning y *transfer learning*, dando lugar a Modelos Neuronales de Lenguaje pre-entrenados con corpora masivo no anotado disponibles en la web como *Wikipedia* y *BookCorpus*, entre otros (Devlin et al., 2018).

El estudio de las dificultades afrontadas por los sistemas de Pregunta-Respuesta recientes es escaso y no hay una forma unificada en cuanto a metodología para el análisis de los errores cometidos por los modelos desde una perspectiva de Procesamiento de Lenguaje Natural con las técnicas mencionadas, ya que actualmente se emplea técnicas de evaluación de Aprendizaje Automático y Deep Learning sobre un conjunto de *test* y una inspección manual de un número reducido de ejemplos como análisis de error: se carece de anotaciones masivas o caracterizaciones de fenómenos lingüísticos de las colecciones (Rogers et al., 2020).

Es decir, los esfuerzos priorizan el uso de manera incremental de técnicas avanzadas a nivel de diseño de Modelos Neuronales de Lenguaje (mayor número de parámetros y datos) para mejorar métricas de evaluación (como la medida F1) sin un análisis en profundidad de la formulación de las preguntas y respuestas y contextualización de más dificultades del lenguaje: presencia de estructuras sintácticas complejas, dificultad de determinar el tipo de respuesta, necesidad de semántica y conocimiento común, lidiar con paráfrasis, omisiones, negaciones o ambigüedades presentes en el lenguaje, entre otros fenómenos lingüísticos.

No se enumeran todos los retos y dificultades que presenta el lenguaje natural de forma global en una colección de Pregunta-Respuesta, ni se analiza de forma particular cómo el modelo de Comprensión Lectora supera cada dificultad ni cada fase: el análisis y contextualización de la pregunta, detección entidades y relaciones en pasajes de los documentos o fuentes de información, lidiar con paráfrasis, errores ortográficos y/o estructuras sintácticas complejas al localizar las fuentes que dan respuesta, para finalmente seleccionar el tipo de respuesta y forma (extracción de la respuesta, selección entre múltiples opciones o sin respuesta) de manera que no contenga información irrelevante y con confianza suficiente para considerarla correcta.

1.1 Objetivos

Tras el estudio del estado del arte del área de Búsqueda de Respuestas y los Modelos Neuronales de Lenguaje, se tiene como objetivo diseñar un modelo predictor de dificultad para detectar qué preguntas y con qué probabilidad serán contestadas erróneamente por un modelo de Pregunta-Respuesta o por una mayoría de ellos. Además, se plantea estudiar si es posible generalizar la predicción en base a la caracterización del lenguaje por su formulación, fragmento y respuesta a extraer permite unificar criterios de dificultad de una colección a otra.

Como punto de partida, se tiene el objetivo de unificar diversas problemáticas y consideraciones heterogéneas en cuanto a las dificultades que plantean las colecciones de Pregunta-Respuesta y retos que superan los Modelos Neuronales de Lenguaje. Por ello, es necesario realizar un estudio de fenómenos lingüísticos de las colecciones de Búsqueda de Respuestas. La comparativa de las colecciones y el análisis por la formulación de sus textos permite caracterizar las colecciones para observar los retos superados o no por los Modelos Neuronales de Lenguaje.

El objetivo inicial de comparar las respuestas otorgadas por una persona (generalmente *crowdworkers*) con las obtenidas con los Modelos Neuronales de Lenguaje de forma automática permite definir el criterio y una anotación por dificultad de forma homogénea para considerar su dificultad de una manera objetiva según distintos niveles y de forma particularizada o glo-

bal. Posteriormente, un método de anotación automático permite detectar preguntas difíciles en menor tiempo y sin esfuerzo manual.

Una vez establecida la metodología de anotación de dificultad, la predicción de dificultad en base a los modelos neuronales permite observar robustez y eficiencia de cada modelo neuronal sobre otros grupos de preguntas con misma o distinta caracterización lingüística para otras colecciones no empleadas en el entrenamiento. También permite evaluar métricas de forma detallada sobre dificultades lingüísticas para todos los Modelos Neuronales de Búsqueda de Respuestas actuales de una forma estandarizada.

1.2 Estructura del documento

En el capítulo 2 se presenta el estudio del estado del arte sobre sistemas de Pregunta-Respuesta con Modelos Neuronales de Lenguaje procedentes de distintas organizaciones: *DrQA* (Universidad de Stanford), *BiDAF* (Instituto Allen de Inteligencia Artificial), *GPT* (OpenAI), *BERT* (Google) y sus principales variaciones como *XLNet*, *RoBERTa* y *T5* (Google). También se recopila estudios que proponen las colecciones de Pregunta-Respuesta: *MCTest*, *CNN Daily Mail*, *SQuAD* (y ejemplos de modificaciones ruidosas), *NewsQA*, *CoQA* y *QuAIL*. En los estudios asociados a cada conjunto se encuentra algún análisis, técnica y enumeración de fenómenos lingüísticos tomados en consideración como punto de partida para el análisis de dificultad.

Adicionalmente, el capítulo 2 incluye los métodos para caracterizar las

preguntas, respuestas y textos de una colección de Pregunta-Respuesta en base a etiquetas lingüísticas obtenidas con técnicas de Procesamiento de Lenguaje Natural: análisis sintáctico, reconocimiento de entidades nombradas y obtención del foco de la pregunta. La ejecución de Modelos Neuronales de Lenguaje permite realizar el estudio de dificultad *a posteriori* en base a los resultados obtenidos con los modelos actuales para nuevas colecciones y modelos.

A partir de los fenómenos lingüísticos, en el capítulo 3 se entra en detalle en la caracterización de dificultad con la combinación de varias perspectivas. En primer lugar, a partir de la formulación del lenguaje de las preguntas, respuestas y contextos se realiza un análisis estadístico global de la colección, tras la obtención de patrones comunes de preguntas. El segundo enfoque consiste en el análisis en base a la ejecución de Modelos Neuronales de Lenguaje sobre una colección para observar dificultades comunes sobre ella.

En el capítulo 4 se expone los planteamientos para anotación de preguntas por su dificultad para un modelo o para la mayoría de ellos y su aplicación sobre unas colecciones concretas en el capítulo 5. Esta anotación de la colección se emplea para el diseño de modelos de predicción de dificultad que a partir de una pregunta, contexto y respuesta, determinen si una pregunta es difícil o no con técnicas de Aprendizaje Automático supervisado, como se detalla en el capítulo 6. También, incluye la evaluación de los modelos de predicción de dificultad con una colección de pregunta-respuesta no empleada en el entrenamiento del modelo.

Tras la descripción de los resultados obtenidos en el análisis de las preguntas y dificultades que presentan la colecciones, en el capítulo 7 se expone las conclusiones y propuestas de mejora sobre los Modelos Neuronales de Lenguaje para Pregunta-Respuesta actuales como trabajo futuro.

En los anexos se dispone de más detalle sobre fases, desarrollos, anotaciones y resultados obtenidos en este trabajo para su consulta y profundización.

Capítulo 2

Estudio del estado del arte

El área de Búsqueda de Respuestas ha evolucionado gracias a dos objetivos principales de los recientes estudios: construcción de nuevas colecciones a gran escala y la aparición de nuevos Modelos Neuronales de Lenguaje entrenados en ellas y más pre-entrenamiento sobre corpus no anotados.

En los estudios contemplados en este trabajo se encuentra consideraciones heterogéneas en cuanto a los fenómenos lingüísticos y tipos de análisis de las colecciones, todos ellos tomados en consideración para el análisis de dificultad preliminar y la selección de métodos utilizados en Procesamiento de Lenguaje Natural y Aprendizaje Automático para su uso posterior en la caracterización lingüística, metodología y predicción de dificultad diferencial de este trabajo.

2.1 Búsqueda de Respuestas

La Búsqueda de Respuestas es un área de estudio que tiene como objetivo obtener información relevante dada una pregunta en lenguaje natural en forma de fragmento, entidad o sintagma que da respuesta en base a su extracción del texto de origen (Pregunta-Respuesta extractivo) o selección entre múltiples opciones denominados distractores (Pregunta-Respuesta multirespuesta), (Martínez-Barco et al., 2014).

En su planteamiento original los sistemas de Pregunta-Respuesta, al igual que los motores de búsqueda del área de Recuperación de la Información, se realiza una primera fase de obtención de una lista ordenada de documentos que son relevantes para la pregunta del usuario para después localizar pasajes válidos o extraer información para generar la respuesta. En el proceso de detectar, extraer y presentar información en un formato para su procesamiento posterior se emplea técnicas de Procesamiento de Lenguaje Natural y las subtareas de Extracción de la Información como el reconocimiento de entidades, extracción de relaciones y detección de eventos y expresiones temporales y espaciales (Sarawagi, 2008). En los más avanzados, también se incluyen métodos de clasificación para determinar si se trata de una pregunta sin respuesta y técnicas de validación (Rodrigo et al., 2018).

Sin embargo, los sistemas de Pregunta-Respuesta se diferencian de los sistemas de Recuperación de la Información por el tipo de lenguaje empleado por los usuarios, generalmente se estructura con formulación sintáctica

correcta y más natural, más similar al que se emplea al conversar o en interfaces de voz, a diferencia del lenguaje típico basado en palabras clave de los motores de búsqueda. Las preguntas pueden clasificarse por el tipo de respuesta que requieren: factuales o hechos concretos (fechas, nombres, cantidades, etc.), de resumen (es necesario localizar información relacionada y resumirla para presentarla) y de opinión (preguntas complejas que recopilan datos y requieren aplicar deducción). Como subtipos de preguntas factuales existen las preguntas sí/no, con términos interrogativos y respuesta de un tipo asociado (quién, cuándo, cómo, dónde, etc.), preguntas indirectas (por ejemplo, empiezan por *me gustaría saber...*) y preguntas de requerimiento (comienzan con la petición del tipo *dime* o *quiero* por ejemplo).

Las respuestas están caracterizadas por su extensión, pueden ser cortas si contestan a preguntas de hechos concretos o factuales, o largas si contestan a preguntas de opinión, de resumen o requerimiento, opcionalmente acompañadas por el origen y localización del documento asociado que da respuesta (Vicedo, 2004). Estos tipos de respuestas pueden ser obtenidas mediante extracción del texto o generadas en el caso de sistemas de Pregunta-Respuesta con técnicas de Generación de Lenguaje y Resumen Automático. En otros casos, si la respuesta no puede ser extraída o generada se da una respuesta válida para notificar que no hay respuesta.

El nivel ideal de conocimiento de estos sistemas debería admitir omisiones en las preguntas, lidiar con semántica y su representación (sinónimos, antónimos, hiperónimos o hipónimos, etc.), contextualizar y sintetizar en las

respuestas conclusiones o decisiones obtenidas de diferentes fuentes, incluso multimedia. Este conocimiento se caracteriza en cuatro niveles acumulativos: de hechos concretos (asociadas las preguntas factuales o *factoid*), explicativo (con bases de conocimiento u ontologías léxico-semánticas como WordNet y más detalladas en el estudio Camacho-Collados and Pilehvar (2018)), modal (permite afrontar preguntas de opinión y resumen, con bases de conocimiento de alto rendimiento) y ámbito o conocimiento general (Vicedo, 2004).

Actualmente, los estudios de sistemas de Pregunta-Respuesta han explotado en la formación de colecciones de preguntas y respuestas para la tarea de Comprensión Lectora. Dichas colecciones permiten evaluar el reto de la detección de preguntas que no es posible contestar o selección de la respuesta entre múltiples opciones, algunas requieren control de contexto y diálogo, o superar la problemática de paráfrasis y similitud semántica textual (Rogers et al., 2020).

Se avanza hacia el enriquecimiento y pre-entrenamiento de los modelos con fuentes de información heterogéneas públicas que conforman corpora no anotado: corpus recopilados de la web, bases de datos documentales, conocimiento de bases de datos, ontologías, fuentes textuales de foros, redes sociales y diálogos (Reddy et al., 2019).

2.1.1 Contextualización histórica

En sus inicios en los años 60 y 70, los sistemas de Pregunta-Respuesta se trataban de interfaces en lenguaje natural a sistemas expertos para dominios restringidos, los más famosos el estudio BASEBALL (el modelo de Green Jr et al. (1961) respondía preguntas sobre la liga de béisbol de USA) y LUNAR (modelo de Woods (1973) sobre análisis geológico de las piedras lunares de las misiones Apollo), a partir de los cuales se empleaba como núcleo una base de datos de conocimiento escrita manualmente por expertos del dominio específico (Martínez-Barco et al., 2014). Otros sistemas conversacionales primitivos como SHRLDU y ELIZA tienen habilidades de pregunta-respuesta en sus simulaciones de comportamiento humano y conversaciones enlatadas (Martínez-Barco et al., 2014).

En su evolución en los años 70 hasta los años 90, los sistemas fundamentaron las fases de procesamiento con técnicas de Recuperación de la Información y Procesamiento de Lenguaje Natural: el análisis de la pregunta (qué tipo de dato se requiere y obtener palabras clave), recuperación de los documentos relevantes (filtrar los documentos con información de las palabras clave de la pregunta), extracción de datos y contextualización, para finalmente realizar la selección o generación de la respuesta (Moldovan et al., 2000).

El ámbito de Búsqueda de Respuesta se formaliza en el año 1999 en las conferencias del Text Retrieval Conference (TREC), y evoluciona con

sistemas enfocados al tratamiento de gran volumen de información (Clarke et al., 2009). Aparecen áreas relacionadas como la búsqueda de respuestas en ficheros de preguntas-respuestas frecuentes o *Frequently Asked Question* (Vicedo, 2004). Posteriormente se formalizan las tareas de comprensión de textos o Comprensión Lectora y búsqueda de respuestas de conocimiento general.

Con el reto de dar respuesta a preguntas de ámbito general, el estudio e investigación para el sistema Watson de IBM Research, victorioso en el concurso de televisión Jeopardy, dio como resultado una nueva arquitectura paralelizable y eficiente en cuanto al número de preguntas contestadas, precisión y tiempo de respuesta (Ferrucci et al., 2010). Las capacidades de pregunta-respuesta de esta versión inicial de Watson consiste en análisis, clasificación y descomposición de la pregunta (con reglas y clasificadores estadísticos), adquisición automática de fuentes de información (con conocimiento somero y profundo con ontologías como DBPedia, WordNet, Yago) y evaluación (con precisión, porcentaje de preguntas contestadas), detección de relaciones y entidades, representación de conocimiento y razonamiento, con múltiples técnicas (Ferrucci et al., 2010). A partir de 20.000 preguntas arbitrarias que interrelacionan una o más entidades, las 30 categorías (del tipo historia, ciencia) y pistas del concurso Jeopardy, se define el concepto de *Lexical Answer Type*: frase nominal que al detectarse de la pregunta completa como unidad independiente la pregunta y tipo de respuesta. Este sistema es la referencia principal para sistemas para apoyo de decisiones y descu-

brimiento de conocimiento en dominios específicos a partir del conocimiento general.

Los estudios recientes de Pregunta-Respuesta se centran en la tarea de Comprensión Lectora con modelos de Deep Learning con arquitectura particular (*DrQA* y *BiDAF*) hasta el uso de modelos pre-entrenados con conjuntos de datos de gran escala y con conocimiento de ámbito general. Con ellos se ha superado la precisión obtenida por un ser humano y rendimiento de otras técnicas de Recuperación de la Información, al igual que en múltiples tareas de Procesamiento de Lenguaje Natural. Así, se observa evolución en los modelos de representación de lenguaje: formas de inicialización con distintos tipos de embedding a nivel de palabra (word2vec, GloVe, ELMo y más detalle en Camacho-Collados and Pilehvar (2018)) o a nivel de carácter o morfológico (FastText, WordPiece, más tipos en Collobert et al. (2011)); el uso de arquitecturas convolucionales (CNN) o secuenciales (LSTM, biLSTM) combinados con más planteamientos y estrategias de Aprendizaje Automático (Collobert et al., 2011) y recientemente, modelos basados en atención (como *Transformer* del estudio Vaswani et al. (2017)) base para los modelos neuronales pre-entrenados a gran escala como GPT (Radford et al., 2018), BERT (Devlin et al., 2018) y T5 (Raffel et al., 2019).

2.1.2 Comprensión Lectora

Como tarea del área de Búsqueda de Respuestas, la Comprensión Lectora define que una máquina comprende un fragmento de texto si para cualquier

pregunta arbitraria que puede ser contestada correctamente por la mayoría de hablantes nativos del idioma la máquina puede proporcionar una cadena de texto con la que estén de acuerdo y no contenga información irrelevante (Jurafsky and Martin, 2009-2021).

El proceso también requiere el análisis de la pregunta (¿se pregunta algo que se referencia explícitamente en el texto, no explícito o la respuesta está en varias oraciones o no se puede contestar?), el análisis de relaciones y entidades en pasajes de los documentos, detectar el tipo de respuesta (un pasaje, una definición, hecho, persona, lugar, etc.) y forma (una respuesta, sin respuesta, multirespuesta) que de forma natural un ser humano realiza en la comprensión del lenguaje natural (Martínez-Barco et al., 2014).

Para que una máquina sea capaz de imitar estos razonamientos, es necesario diseñar técnicamente un sistema o modelo de lenguaje en el ámbito de conocimiento en el que se adapta el sistema (dominio específico o general, único dominio o múltiples), realizar su enriquecimiento y diseño de la estrategia o entrenamiento del modelo de análisis y representación de la pregunta para la obtención, formación o selección de la respuesta. Para la evaluación de su rendimiento, un test de lectura o examen de comprensión lectora está compuesto por un texto principal y un conjunto de preguntas con respuesta única o múltiple, de complejidad variada para comprobar la capacidad de entendimiento, inferencia y conocimiento general alcanzado por el lector o por un modelo de Inteligencia Artificial mediante la comparación de sus respuestas con la correcta de forma estandarizada (Clark and Etzioni, 2016).

La nueva forma de evaluación de los Modelos Neuronales de Lenguaje para Pregunta-Respuesta consiste en la ejecución del modelo sobre preguntas no vistas en el entrenamiento (conjunto de test) para obtener la predicción de la respuesta y el cálculo de métricas basadas en exactitud (accuracy), precisión y cobertura, como la puntuación F1, empleando la perspectiva de evaluación de Aprendizaje Automático y Deep Learning (Rodrigo and Peñas, 2017). Además, la prioridad de los estudios es el diseño de nuevas arquitecturas de Deep Learning para representación del lenguaje que superen en eficiencia a los estudios previos, (Ruder, 2018a), empleando parámetros pre-entrenados de dichos modelos y su especialización (Ruder et al., 2019).

2.2 Colecciones de Pregunta-Respuesta

Para realizar la evaluación de los sistemas de Pregunta-Respuesta se forman exámenes de comprensión lectora (Clark and Etzioni, 2016) y más colecciones basados en corpus de documentos, fragmentos de ellos y la elaboración de preguntas y respuestas para cada documento o fragmento del mismo, centrandose así en la tarea de Comprensión Lectora de los sistemas de Pregunta-Respuesta. En mayor parte de los conjuntos de datos recientes se omite la selección de los fragmentos y documentos candidatos a contener la respuesta a la pregunta de un usuario. Dicha evaluación adicional reta a los sistemas para ver su capacidad de automatización de la indagación, localización y lectura de los documentos que realiza un usuario, comportamiento típico en el

uso de los motores de búsqueda web y sistemas de indexación de contenidos.

Se observa que los textos a partir de los cuales se formulan preguntas tienen origen diverso: desde los exámenes de Comprensión Lectora (*MCTest*, ver 2.2.1, y *RACE*), noticias (*CNN Daily Mail*, ver 2.2.2), artículos de *Wikipedia* (caso de *SQuAD* (Rajpurkar et al., 2016)); hasta en foros (*Quora*), buscadores (*Natural Questions*), conversaciones (*CoQA*, en detalle en 2.2.6) y combinación de múltiples dominios (Rogers et al., 2020) y más recursos digitales masivos. Todas ellas tienen en común que están formuladas en inglés.

A continuación se describe las colecciones de Pregunta-Respuesta más utilizadas actualmente. A la hora de caracterizar una colección es necesario contemplar qué tareas dentro de Pregunta-Respuesta evalúan y si incluye la búsqueda en varios documentos o uno solo. Posteriormente, el objetivo a resolver varía si hay que devolver un fragmento del texto o extraer la respuesta del fragmento o determinar si es no contestable (casos de *SQuAD* y *NewsQA* (Trischler et al., 2017) descritas en 2.2.3 y 2.2.5, respectivamente) o seleccionar la respuesta correcta entre varias opciones (como *RACE*, detallada en 2.2.4).

2.2.1 MCTest

Microsoft Research publica este conjunto de historias y preguntas asociadas en inglés en el estudio Richardson et al. (2013) con el objetivo de comparar la tarea de pregunta-respuesta abierta mediante test de comprensión lectora.

El conjunto de datos se creó mediante crowd-sourcing con Amazon Mechanical Turk y consiste en 660 historias de 150 a 300 palabras y 2.640 preguntas. La evaluación se realiza con la selección de la correcta de entre cuatro respuestas, evitando las problemáticas de ambigüedad y ranking de oraciones sobre un documento.

El conjunto de datos se divide en dos por su objetivo: *MC160* (para entrenamiento) y *MC500* (para test). Para el primero se anotan distintos niveles de dificultad (por curso académico cursos 1º a 4º) y se revisaron manualmente errores ortográficos contrastando con lexicones. Posteriormente, para la generación de *MC500*, se automatiza el proceso para permitir escalabilidad: para corregir errores gramaticales y revisión con una gramática.

Esta colección no se ha empleado para el entrenamiento de Modelos Neuronales de Lenguaje debido a su menor tamaño, pero supone un punto de partida para la comparación.

2.2.2 CNN Daily Mail

Google DeepMind y la Universidad de Oxford crearon los conjuntos de datos *CNN* y *Daily Mail* empleando artículos de dichas fuentes en inglés, respectivamente. El primero consta de 90.266 artículos y unas 380 mil preguntas, el segundo 197 mil documentos y 879 mil preguntas. Para la generación de las preguntas se emplea un método automático: se extrae una frase del artículo sin alguna palabra de entidad que se encuentra en el documento de origen y

es candidato a ser la respuesta.

El análisis realizado del dataset sobre 100 ejemplos tomados aleatoriamente (Chen et al., 2016) permite extraer las consideraciones lingüísticas de la figura 2.1 como primera caracterización de las preguntas y respuestas y retos que afrontar con los modelos.

- Presencia de los mismos términos en la pregunta y el fragmento que contiene la entidad de respuesta (*exact match*). 13 %.
- Paráfrasis a nivel de oraciones entre la pregunta y la respuesta. 41 %.
- Presencia de pistas parciales con solapamiento de palabra o concepto entre la pregunta y la respuesta, aunque por la semántica no haya asociación directa. 19 %.
- Se requiere el procesamiento de múltiples oraciones. 2 %.
- Se encuentra errores en coreferencias, de manera que la pregunta no es contestable. 8 %.
- Ambigüedades o muy complejas, aquellas en las que un ser humano tendría duda. 17 %.

Se observa que los modelos basados en modelos secuenciales (encoder con RNN y/o LSTM) y de atención superan a los modelos que emplean solamente representación (ventanas semánticas, distancias de palabras) y/o clasificación en los casos enmarcados en las primeras cuatro consideraciones

Category	Question	Passage
Exact Match	*it 's clear @entity0 is leaning toward* **@placeholder** , says an expert who monitors @entity0	... @entity116 , who follows @entity0 's operations and propaganda closely , recently told @entity3 , *it 's clear @entity0 is leaning toward* **@entity60** in terms of doctrine , ideology and an emphasis on holding territory after operations . ...
Paraphrase	**@placeholder** says he understands why @entity0 wo n't play at his tournament	... @entity0 called me personally to let me know that he would n't be playing here at @entity23 , " **@entity3** said on his @entity21 event 's website . ...
Partial clue	a tv movie based on @entity2 's book **@placeholder** casts a @entity76 actor as @entity5	... to @entity12 @entity2 professed that his **@entity11** is not a religious book . ...
Multiple sent.	he 's doing a his - and - her duet all by himself , @entity6 said of **@placeholder**	... we got some groundbreaking performances , here too , tonight , @entity6 said . we got **@entity17** , who will be doing some musical performances . he 's doing a his - and - her duet all by himself . ...
Coref. Error	rapper **@placeholder** " disgusted , " cancels upcoming show for @entity280	... with hip - hop star **@entity246** saying on @entity247 that he was canceling an upcoming show for the @entity249 . ... (but @entity249 = @entity280 = SAEs)
Hard	pilot error and snow were reasons stated for **@placeholder** plane crash	... a small aircraft carrying **@entity5** , @entity6 and @entity7 the @entity12 @entity3 crashed a few miles from @entity9 , near @entity10 , @entity11 . ...

Tabla 2.1: *Análisis de consideraciones lingüísticas de pregunta y respuesta en CNN Daily Mail datasets (Chen et al., 2016).*

lingüísticas, mientras que son más débiles para los casos de coreferencias y ambigüedades.

Este conjunto no permite evaluar la Comprensión Lectora de pregunta-respuesta a través de inferencia y semántica avanzada, pero permitió entrenar los primeros sistemas basados en modelos neuronales por su mayor tamaño, como *BiDAF*; y para componer nuevas colecciones de Pregunta-Respuesta: *NewsQA* (Trischler et al., 2017).

2.2.3 SQuAD

Stanford Question Answering Dataset (*SQuAD*) es una colección de pregunta-respuesta en inglés creada por crowdworkers ante la necesidad de tener un corpus de gran tamaño para estudiar nuevos modelos de comprensión lectora. Consta de 107.785 pares preguntas y su correspondiente respuesta (fragmento de oración o segmento del texto del artículo a extraer) creados a partir de 536 artículos de Wikipedia.

En el estudio Rajpurkar et al. (2016) se incluye el análisis de las propiedades del conjunto de desarrollo: tipos de respuestas, dificultad de las preguntas por la necesidad de lógica e inferencia, grado de divergencia sintáctica entre pregunta y respuesta. Las respuestas a extraer no se limita a entidades nombradas, y puede incluir sintagmas nominales complejos.

Para entender la dificultad de las preguntas en el estudio se caracteriza la diversidad del tipo de respuestas, los tipos de razonamientos necesarios para contestarlas y el grado de divergencia entre la pregunta y las posibles oraciones de respuesta. En concreto, se tiene las siguientes consideraciones tras el análisis de 192 ejemplos manualmente.

- Los tipos de respuestas pueden ser numéricas o no numéricas. Las segundas se categorizan con el uso de un analizador sintáctico, dividiéndose en función de detección de persona, localización y otras entidades por su etiquetado.

- El razonamiento necesario para responder las preguntas son los siguientes: variación léxica por sinonimia, variación léxica por conocimiento general, variación sintáctica, razonamiento en varias oraciones y coreferencias, ambiguedad.

En la segunda versión del conjunto de datos, el estudio Rajpurkar et al. (2018) añade 53.775 preguntas que no pueden ser contestadas y son similares a las que sí para avanzar en el reto de determinar cuando no es posible dar una respuesta, bien por no estar contenida en el párrafo a pesar de tratar la misma temática (son preguntas relevantes) o el tipo de respuesta correcto para la pregunta no aparece como fragmento del párrafo.

Las preguntas de esta segunda versión han sido generadas por crowdworkers que observan las preguntas contestables y el párrafo y generan la no contestable similar a las otras, por ejemplo por negación, uso de antonimia, cambio de entidad, exclusión mutua o pregunta de una condición imposible (ver tabla 2.2). También se han entrenado modelos basados en TF-IDF (con solapamiento de palabras) y reglas para generación de preguntas no contestables.

2.2.3.1 Modificación ruidosa de SQuAD

Para la caracterización de la robustez de los modelos neuronales de lenguaje el estudio Aspillaga et al. (2020) elabora una versión del dataset de desarrollo de *SQuAD 1.1* con distintas estrategias: modificación de fragmentos

Reasoning	Description	Example	Percentage
Negation	Negation word inserted or removed.	Sentence: "*Several hospital pharmacies have decided to outsource high risk preparations . . .* " Question: "*What types of pharmacy functions have* ***never*** *been outsourced?*"	9%
Antonym	Antonym used.	S: "*the extinction of the dinosaurs. . . allowed the tropical rainforest to spread out across the continent.*" Q: "*The extinction of what led to the* ***decline*** *of rainforests?*"	20%
Entity Swap	Entity, number, or date replaced with other entity, number, or date.	S: "*These values are much greater than the 9–88 cm as projected . . . in its Third Assessment Report.*" Q: "*What was the projection of sea level increases in the* ***fourth assessment report****?*"	21%
Mutual Exclusion	Word or phrase is mutually exclusive with something for which an answer is present.	S: "*BSkyB. . . waiv[ed] the charge for subscribers whose package included two or more premium channels.*" Q: "*What service did BSkyB* ***give away for free unconditionally****?*"	15%
Impossible Condition	Asks for condition that is not satisfied by anything in the paragraph.	S: "*Union forces left Jacksonville and confronted a Confederate Army at the Battle of Olustee. . . Union forces then retreated to Jacksonville and held the city for the remainder of the war.*" Q: "*After what battle did Union forces leave Jacksonville* ***for good****?*"	4%
Other Neutral	Other cases where the paragraph does not imply any answer.	S: "*Schuenemann et al. concluded in 2011 that the Black Death. . . was caused by a variant of Y. pestis. . .* " Q: "*Who* ***discovered*** *Y. pestis?*"	24%
Answerable	Question is answerable (i.e. dataset noise).		7%

Tabla 2.2: *Tipos de preguntas no contestables y proporción de preguntas de una muestra de 100 ejemplos (Rajpurkar et al., 2018)*

de texto versus modificaciones a nivel de palabra. Estas modificaciones permiten añadir información adicional o errónea y simular errores ortográficos presentes en las colecciones, como se ejemplifica a continuación.

En la modificación de los fragentos de texto, se añade una oración adversa creada a partir de una de las cuatro de las siguientes técnicas: *AddOneSent*, *AddSent*, *AddAny*, *AddCommon*.

- *AddOneSent* toma la pregunta, se reemplaza sus sustantivos y adjetivos por antónimos. En el caso de entidades nombradas, por aquella similar en el word embedding GloVe, y se reescribe de forma declarativa como

se muestra en la figura 2.1.

- Si se añade más de una oración creada con la técnica anterior, se crea más confusión obteniendo la técnica *AddSent*.

- Si al generar el ejemplo se sustituyen palabras por otras aleatorias de un conjunto seleccionado de 20 palabras contenidas en la pregunta y más palabras comunes que minimizan la confianza del modelo en la respuesta correcta, se emplea la técnica *AddAny*.

- Si solo se emplean las palabras comunes en la técnica anterior, se tiene *AddCommon*.

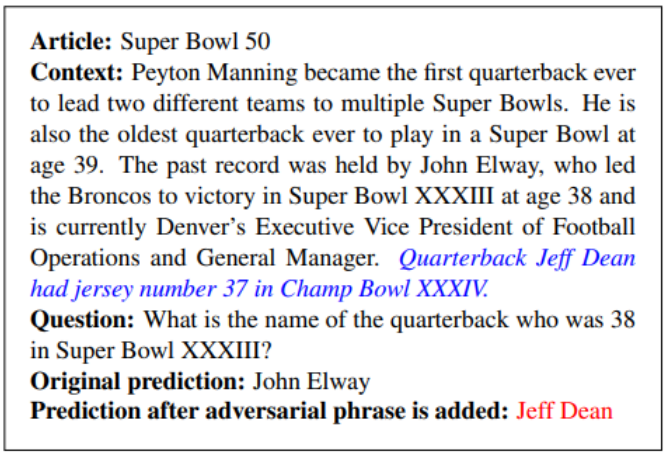


Figura 2.1: *Ejemplo de AddOneSent, en azul oración adversa añadida, en rojo fallos.*

Sobre el texto del pasaje se introducen modificaciones a nivel de palabra, si afecta a la respuesta también se modifica, con distintas técnicas de ruido:

Natural Noise, *Swap Noise*, *Middle Random Noise*, *Fully Random Noise* y *Keyboard Typo Noise*.

- En *Natural Noise* se sustituyen palabras por aquellas con errores ortográficos típicos cometidos por usuarios en plataformas web.
- En *Swap Noise* se permutan caracteres consecutivos de la palabra elegidos aleatoriamente.
- En *Middle Random Noise* se toma los caracteres de la palabra y se ordenan aleatoriamente, dejando fijos el primero y último caracter de la palabra original.
- *Fully Random Noise* modifica todos los caracteres de la palabra aleatoriamente.
- En *Keyboard Typo Noise* se reemplazan caracteres por su adjunto en el teclado inglés.

2.2.4 RACE

La colección *ReAding Comprehension dataset from Examinations* (Lai et al., 2017) está formada a partir de exámenes reales de Comprensión Lectora en inglés relizados por estudiantes de 12 a 18 años, creados por los evaluadores (expertos del dominio). Recopila 27.933 textos, 97.687 preguntas y respuestas asociadas. Por su planteamiento esta colección permite evaluar el modo

de pregunta-respuesta múltiple: para cada pregunta se disponen cuatro opciones de respuesta en las que solo una de ellas es correcta y las demás son distractores que varían en forma en función de la dificultad de la pregunta, ver el ejemplo 2.2.

Al responder se requiere capacidad de razonamiento a nivel de análisis en detalle y global, capacidad de síntesis y de entendimiento de opinión del texto ofrecido en variedad de formas, estilos (artículos, noticias, historias) y temáticas (filosofía, biográficos, etc.) (Lai et al., 2017). Otras dificultades a nivel lingüístico observadas en el estudio Lai et al. (2017) a partir del análisis de 100 ejemplos son: la coincidencia entre palabras versus paráfrasis, razonamiento de una oración versus múltiple y la capacidad de determinar si la pregunta no puede ser respondida por información insuficiente o es ambigua o necesidad de conocimiento general.

Evidence: "...Many people optimistically thought industry awards for better equipment would stimulate the production of quieter appliances. It was even suggested that noise from building sites could be alleviated ..."

Question: What was the author's attitude towards the industry awards for quieter?

Options: A.suspicious B.positive C.enthusiastic D.indifferent

Figura 2.2: *Ejemplo de RACE evidencia de opinión del autor del texto, pregunta y múltiples respuestas (Lai et al., 2017)*

Esta colección supera las limitaciones de las colecciones previas, además de tener gran tamaño, consta de mayor naturalidad y variabilidad de lenguaje al no seguir el planteamiento de pregunta-respuesta extractivo, es decir, no

se limita a acotar la respuesta dentro del fragmento ya que la respuesta ha sido creada para evaluar la Comprensión Lectora de los estudiantes.

Supone mayor dificultad para los Modelos Neuronales del Lenguaje basados en *Transformer* ya que se obtienen menores valores en términos de exactitud y F1-score, como se puede observar en la figura 2.3 en comparación con las figuras de evaluación con *SQuAD* (figuras 2.13, 2.16) y *GLUE*, (figura 2.15).

RACE	Accuracy	Middle	High
GPT [28]	59.0	62.9	57.4
BERT [25]	72.0	76.6	70.1
BERT+DCMN* [38]	74.1	79.5	71.8
RoBERTa [21]	83.2	86.5	81.8
XLNet	**85.4**	**88.6**	**84.0**

Figura 2.3: *Evaluación con RACE para GPT, BERT, RoBERTa y XLNet Yang et al. (2019)*

2.2.5 NewsQA

Esta colección presenta mayor reto para los Modelos Neuronales de Lenguaje para la tarea de Comprensión Lectora gracias a sus 119.633 preguntas creadas por *crowdworkers* sobre las noticias de *CNN* (Chen et al., 2016), ya que las preguntas se forman a partir de la curiosidad de los crodworkers al ver el título y entradilla de la noticia, sin disponer de la noticia completa. Los tipos de razonamientos necesarios son paráfrasis, inferencia, capacidad de síntesis y lidiar ante ambigüedades, además del solapamiento entre pregunta

y respuesta, como se puede observar en la tabla 2.3.

Reasoning	Example	Proportion (%) *NewsQA*	*SQuAD*
Word Matching	Q: **When were** the **findings published**? S: Both sets of research **findings were published Thursday**...	32.7	39.8
Paraphrasing	Q: **Who** is the **struggle between** in Rwanda? S: The **struggle pits ethnic Tutsis**, supported by Rwanda, **against ethnic Hutu**, backed by Congo.	27.0	34.3
Inference	Q: **Who** drew **inspiration** from **presidents**? S: **Rudy Ruiz** says the lives of US **presidents** can make them **positive role models** for students.	13.2	8.6
Synthesis	Q: **Where** is **Brittanee Drexel** from? S: The mother of a 17-year-old **Rochester**, **New York** high school student ... says she did not give her daughter permission to go on the trip. **Brittanee** Marie **Drexel**'s mom says...	20.7	11.9
Ambiguous/Insufficient	Q: **Whose mother** is **moving** to the White House? S: ... **Barack Obama's mother-in-law**, Marian Robinson, will **join** the Obamas at the **family's private quarters** at 1600 Pennsylvania Avenue. [Michelle is never mentioned]	6.4	5.4

Tabla 2.3: *Tipos de razonamientos y proporción de preguntas de una muestra de 1000 ejemplos de SQuAD y NewsQA (Trischler et al., 2017)*

Las respuestas a las preguntas son fragmentos de textos de longitud arbitraria e incluye preguntas no contestables (*null*), permitiendo evaluar la tarea de Pregunta-Respuesta extractivo no multirespuesta. Se obtienen con tres personas distintas que contestan cada pregunta y dos personas que validan las respuestas posteriormente, obteniendo en situaciones más de un fragmento considerado como válido.

El estudio Trischler et al. (2017) compara no solo los tipos de razonamientos mencionados, sino la variedad lingüística de esta colección con respecto a *SQuAD* con técnicas de Procesamiento de Lenguaje Natural para la obtención de etiquetas sintácticas y reconocimiento de entidades sobre todas las respuestas con *Stanford CoreNLP* (Qi et al., 2018).

Answer type	Percentage	Example
Date	8.9%	19 October 1512
Other Numeric	10.9%	12
Person	12.9%	Thomas Coke
Location	4.4%	Germany
Other Entity	15.3%	ABC Sports
Common Noun Phrase	31.8%	property damage
Adjective Phrase	3.9%	second-largest
Verb Phrase	5.5%	returned to Earth
Clause	3.7%	to avoid trivialization
Other	2.7%	quietly

Tabla 2.4: *Tipos de respuestas de SQuAD (Rajpurkar et al., 2016)*

Comparando las tablas 2.4 y 2.5 se observa que *NewsQA* tiene mayor proporción de respuestas basadas en oraciones subordinadas (*clause phrase*), localizaciones, personas, y otros tipos sintácticos de respuestas, además de respuestas con sintagmas preposicionales, pero menor variedad en otros tipos de entidades.

Answer type	Example	Proportion (%)
Date/Time	March 12, 2008	2.9
Numeric	24.3 million	9.8
Person	Ludwig van Beethoven	14.8
Location	Torrance, California	7.8
Other Entity	Pew Hispanic Center	5.8
Common Noun Phr.	federal prosecutors	22.2
Adjective Phr.	5-hour	1.9
Verb Phr.	suffered minor damage	1.4
Clause Phr.	trampling on human rights	18.3
Prepositional Phr.	in the attack	3.8
Other	nearly half	11.2

Tabla 2.5: *Tipos de respuestas de NewsQA (Trischler et al., 2017)*

2.2.6 CoQA

La colección *Conversational Question Answering* del estudio Reddy et al. (2019) es novedosa por su formato y plantea la evaluación de los modelos ante coreferencias, pragmática, y contextualización conversacional en los sistemas de Pregunta-Respuesta.

Gracias a crowd-sourcing con Amazon Mechanical Turk, la Universidad de Stanford recopila 8.000 conversaciones en inglés de siete dominios distintos para la creación del corpus de 127.000 preguntas y sus respuestas (extraídas de la conversación) con su evidencia (oración completa de la conversación).

La colección se crea gracias a dos anotadores para cada conversación en la plataforma interactiva creada con ParlAI MTurk API. Uno de ellos crea la pregunta y el otro contesta por turnos; también se cuenta con validación del compañero ante respuestas imprecisas o incorrectas. Además, el reto principal que se les propone es evitar contestar con palabras exactas del fragmento de la conversación donde se encuentra la evidencia de la respuesta.

Así, se consigue que un 33.2 % de respuestas sean abiertas y con lenguaje más variado y no solo se evalúe pregunta-respuesta extractivo (66.8 % restante), es decir, se evita que la respuesta se obtenga por extracción de un fragmento del texto. Las preguntas tienen un lenguaje más natural, y como en una conversación, dependen del orden en el que son formuladas y respondidas para contextualizar información posteriormente, cambiando de entidad de foco y más información, como se puede observar en la figura 2.4.

The Virginia governor's race, billed as the marquee battle of an otherwise anticlimactic 2013 election cycle, is shaping up to be a foregone conclusion. Democrat Terry McAuliffe, the longtime political fixer and moneyman, hasn't trailed in a poll since May. Barring a political miracle, Republican Ken Cuccinelli will be delivering a concession speech on Tuesday evening in Richmond. In recent ...

Q_1: What are the candidates **running** for?
A_1: Governor
R_1: The Virginia governor's race

Q_2: **Where**?
A_2: Virginia
R_2: The Virginia governor's race

Q_3: Who is the democratic candidate?
A_3: **Terry McAuliffe**
R_3: Democrat Terry McAuliffe

Q_4: Who is **his** opponent?
A_4: **Ken Cuccinelli**
R_4 Republican Ken Cuccinelli

Q_5: What party does **he** belong to?
A_5: Republican
R_5: Republican Ken Cuccinelli

Q_6: Which of **them** is winning?
A_6: Terry McAuliffe
R_6: Democrat Terry McAuliffe, the longtime political fixer and moneyman, hasn't trailed in a poll since May

Figura 2.4: *Ejemplo de CoQA, en color coreferencias de la conversación (Reddy et al., 2019)*

Esta colección recopila múltiples dominios (ver tabla 2.6): historias para niños de *MCTest*, literatura, exámenes de cursos académicos de *RACE*, noticias de CNN, artículos de Wikipedia, Reddit y artículos de ciencia de *AI2Science Questions*. Para crear conversaciones se han preprocesado los pasajes de las colecciones y seleccionado aquellos en los que aparecen múlti-

ples entidades, eventos y oraciones con pronombres y sintagmas nominales, empleando Stanford CoreNLP (Manning et al., 2014).

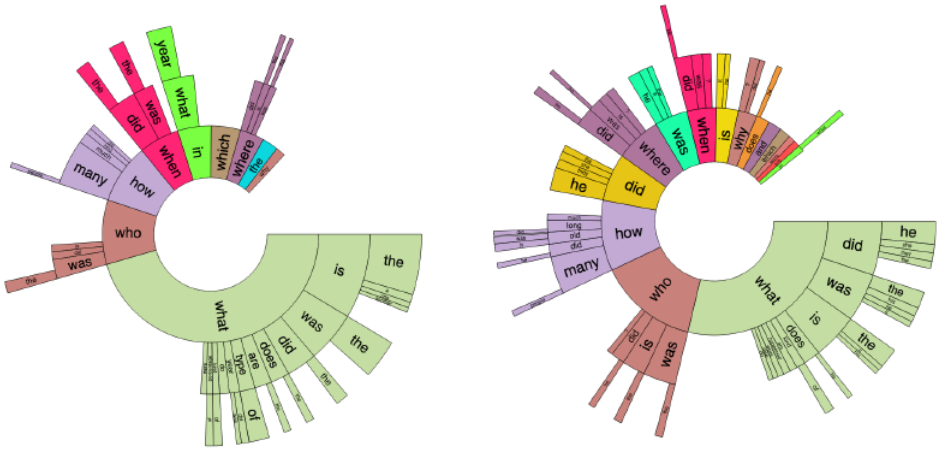
Domain	#Passages	#Q/A pairs	Passage length	#Turns per passage
In-domain				
Children's Sto.	750	10.5k	211	14.0
Literature	1,815	25.5k	284	15.6
Mid/High Sch.	1,911	28.6k	306	15.0
News	1,902	28.7k	268	15.1
Wikipedia	1,821	28.0k	245	15.4
Out-of-domain				
Reddit	100	1.7k	361	16.6
Science	100	1.5k	251	15.3
Total	8,399	127k	271	15.2

Tabla 2.6: *Dominios de CoQA (Reddy et al., 2019)*

El estudio Reddy et al. (2019) incluye un análisis de la colección y comparación con la colección de SQuAD 2.0 (Rajpurkar et al., 2018). En dicho estudio se obtiene la longitud media de la pregunta, respuesta y pasaje, también de la distribución de trigramas de inicio de las preguntas. Al permitirse en *CoQA* preguntas más naturales se observa que las preguntas tienen menor longitud en media, 5.5 palabras, versus a las 10 palabras en media de SQuAD. Las respuestas tienen 3.2 palabras en media en SQuAD y 2.7 en CoQA.

En la figura 2.5 se observa que cerca de la mitad de las preguntas de SQuAD comienzan con *what*, mientras que las preguntas de CoQA son más variadas, tanto en tipo de preguntas (como se puede ver en los sectores de

las partículas *who*, *how*, *where*), como en lenguaje natural por la presencia de los sectores *did*, *was*, *is*, *does* y *and* en CoQA y no SQuAD.



(a) SQuAD (b) CoQA

Figura 2.5: *Distribución de trigramas de inicio de SQuAD 2.0 y CoQA (Reddy et al., 2019)*

Para el análisis de fenómenos lingüísticos se analizan 150 ejemplos del conjunto de desarrollo y test. Se consideran los fenómenos de coincidencia léxica (29.8 % de los casos) versus paráfrasis debido a la presencia de sinónimos, antónimos, hiperonimia, hiponimia y negación (43 % de los casos) y pragmática (27.2 %); fenómenos ejemplificados en la tabla 2.7.

En los modelos evaluados en el estudio se incluyen modelos secuenciales basados en LSTMs bidireccionales, con arquitectura *seq2seq*, el modelo DrQA original y aumentado, comparándose con la métrica F1 obtenido por el humano. Se observa que el modelo neuronal BERT supera las métricas obtenidas en el estudio.

Phenomenon	Example	Percentage
Relationship between a question and its passage		
Lexical match	Q: Who had to rescue her? A: the coast guard R: Outen was rescued by the coast guard	29.8%
Paraphrasing	Q: Did the wild dog **approach**? A: Yes R: he **drew** cautiously **closer**	43.0%
Pragmatics	Q: Is Joey a male or female? A: Male R: it looked like a stick man so she kept **him**. She named her new noodle friend Joey	27.2%
Relationship between a question and its conversation history		
No coref.	Q: What is IFL?	30.5%
Explicit coref.	Q: Who had Bashti forgotten? A: the puppy Q: What was **his** name?	49.7%
Implicit coref.	Q: When will Sirisena be sworn in? A: 6 p.m local time Q: **Where**?	19.8%

Tabla 2.7: *Fenómenos lingüísticos de una muestra de 150 ejemplos de CoQA (Reddy et al., 2019)*

2.2.7 QuAIL

Tras la explosión de colecciones de Pregunta-Respuesta para Comprensión Lectora, el estudio Rogers et al. (2020) propone evaluar más técnicas de entendimiento del lenguaje y razonamiento para decidir si la información está presente en la colección de distintas fuentes textuales y estudiar la confianza que se obtiene para información no disponible en el conjunto de entrenamiento, conocimiento general versus específico, de forma balanceada en número de tipos de preguntas, razonamientos y con anotaciones por ejemplo. Todo ello en forma de Pregunta-Respuesta múltiple en inglés.

Para ello, se crea *Question Answering for Artificial IntelLigence* (*QuAIL*),

una colección que contiene 15.000 preguntas de 800 textos provenientes de fuentes digitales bajo licencia abierta *Creative Commons* de cuatro dominios distintos: libros de ficción, noticias, blogs e historias de usuarios de *Quora*. Estos dominios se diferencian en que los textos de noticias y blogs requieren información que esté presente en corpora masivo y de conocimiento general, mientras que los dominios de las noticias e historias de *Quora* son autocontenidos en eventos e información.

Así, la evaluación sobre cada partición por anotación disponible permite evaluar las dificultades que supera un modelo de Pregunta-Respuesta. Además, a diferencia de otras colecciones de Pregunta-Respuesta como *SQuAD*, esta colección permite evaluar el rendimiento sobre textos de entre 300 y 350 palabras, y consta de mayor variedad en el tipo de preguntas y de forma balanceada en cuanto a tipos de preguntas: factuales, preguntas que requieren conocimiento general y preguntas sin respuesta.

Las preguntas factuales pueden ser contestadas a partir de razonamiento verbal (coreferencias), detección de eventos o identificación de entidades. En las preguntas que requieren conocimiento general se incluyen relaciones de causalidad o vinculación a otras fuentes, inferencia, requieren análisis de opinión o de consecuencias tras hechos narrados o duranate la narración. Las preguntas sin respuesta no pueden ser contestadas por la información disponible en el texto y el conocimiento adquirido de otras fuentes no permiten tampoco dar respuesta.

Para garantizar la corrección y calidad de la colección su creación se lleva a cabo en distintas fases; primero con procesos de creación manual por crowd-founding, una segunda revisión manual por estudiantes, un filtrado y modificación posterior automática, y finalmente una revisión por lingüistas que garantice presencia de paráfrasis y de ejemplos adversos. Para conseguir el número balanceado de ejemplos se exige al anotador que lee el texto la creación de una pregunta de cada tipo, además de marcar las respuestas correctas o plausibles en el formato multi-respuesta. En esta metodología se ejemplifica la dificultad que tiene una persona también para lidiar con múltiples dominios y fuentes de información con confianza y certidumbre.

Adicionalmente, mediante crowd-sourcing se obtiene la exactitud para la evaluación de la colección por personas, ver la tabla 2.8, y se compara con la obtenida en distintos planteamientos técnicos para Comprensión Lectora y similitud semántica textual.

Question type	All questions	Text+ Unanswerable	World knowledge
Temp. order	0.66	0.67	–
Coreference	0.70	0.79	–
Factual	0.75	0.82	–
Causality	0.76	–	0.86
Subsequent	0.53	–	0.62
Duration	0.32	–	0.37
Properties	0.67	–	0.78
Beliefs	0.62	–	0.85
Unanswerable	0.25	0.83	–
Total	0.60	0.78	0.70

Tabla 2.8: *Tipos de pregunta y exactitud obtenida en la evaluación del rendimiento humano en QuAIL (Rogers et al., 2020)*

2.3 Modelos Neuronales de Lenguaje en Búsqueda de Respuestas

En su forma primitiva, un modelo de lenguaje tiene como objetivo predicir una palabra a partir de las secuencias de palabras anteriores (Jurafsky and Martin, 2009-2021). Para ello, realiza una representación a partir de constituyentes o rasgos tomados en base a un corpus o dominio específico. Con enfoque estadístico se obtiene la importancia de los rasgos de forma numérica a nivel local (en un contexto, frase o documento) o global (en todo el corpus) de tal forma que la semántica de una palabra queda influenciada por las palabras que la acompaña, planteamiento conocido como la hipótesis distribucional (*You shall know a word by the company it keeps (Firth, J. R. 1957)*).

Las técnicas empleadas para la representación del lenguaje y su significado han avanzado desde los métodos de representación lógica (bases de conocimiento, reglas y gramáticas), modelos vectoriales (funciones de pesado, modelo de espacio vectorial, TF-IDF), modelos probabilísticos (n-gramas, *Hidden Markov Models* y LDA), indexación y latencia semántica (factorización matricial), modelos estadísticos de Aprendizaje Automático (regresión, clasificación, agrupación no supervisada) hasta los modelos de redes neuronales (convolucionales, secuenciales, *embeddings* (Camacho-Collados and Pilehvar, 2018)) para representación de palabras y oraciones (Youngy et al., 2018). También, se ha aumentado la capacidad de representación de secuen-

cias largas de lenguaje y se ha mejorado la contextualización y atención entre fragmentos y oraciones, especialmente con el pre-entrenamiento de los modelos basados en *Transformer* (Ruder, 2018b).

Gracias al avance de las arquitecturas y pre-entrenamiento de los Modelos Neuronales se han propuesto múltiples sistemas de Pregunta-Respuesta con arquitecturas de Deep Learning entrenados con colecciones de gran tamaño gracias a la web y crowd-sourcing. Los estudios recientes reflejan que se obtienen resultados cercanos al rendimiento humano en términos de precisión (es preferible no obtener respuesta versus a obtener una respuesta incorrecta), cobertura (poder contestar el mayor número de preguntas de forma correcta) y corrección y relevancia de la respuesta (fragmento localizado correctamente y con las palabras necesarias). La métrica de evaluación más extendida en pregunta respuesta extractivo es la media arónica de precisión y cobertura, denominada F1.

En esta sección se describe los primeros estudios DrQA y BiDAF basados en componentes de representación de palabras (*word embedding*), técnicas de Recuperación de la Información y Procesamiento de Lenguaje Natural (TFIDF, análisis sintáctico, reconocimiento de entidades) (Jurafsky and Martin, 2009-2021) y de oraciones (como modelos secuenciales LSTM (Collobert et al., 2011)) y el avance hacia el uso de Modelos Neuronales de Lenguaje pre-entrenados basados en la arquitectura de atención *Transformer* (Vaswani et al., 2017) tales como GPT (Radford et al., 2018), BERT (Devlin et al., 2018) y T5 (Raffel et al., 2019) como estudio del estado del arte.

2.3.1 DrQA

El estudio conjunto de la Universidad de Stanford y Facebook (Chen et al., 2017) emplea la perspectiva de Recuperación de la Información y modelo neuronal de *deep learning* usando como base de conocimiento los artículos de Wikipedia (Manning, 2020).

En primer lugar realiza la recuperación de documentos con un componente de espacio vectorial basado en TF-IDF para obtener los artículos relevantes dada una pregunta.

En segundo lugar, se aplica un modelo neuronal recurrente para detectar el fragmento correcto de respuesta. Este modelo recibe como entrada el pasaje y lo representa con LSTMs bidireccionales a partir del word embedding GloVe, etiquetas gramaticales (*POSTags*) y reconocimiento de entidades (*NER*). Junto con la pregunta se emplea una combinación lineal de alineamiento y atención pregunta-pasaje. La representación de la pregunta es en base a una combinación lineal de los vectores de la palabra, donde el peso es la importancia de la palabra en la pregunta.

Por último, se realiza la predicción del token de inicio y final para seleccionar el fragmento de respuesta, entrenando dos clasificadores multiclase.

2.3.2 BiDAF

El trabajo propuesto en Seo et al. (2016) por el instituto Allen de Inteligencia Artificial está basado en un modelo neuronal jerárquico que representa el contexto de un párrafo con distinta granularidad (a nivel de caracter, palabra y embedding de la frase con LSTMs), seguido de modelos de atención (*Attention Flow layer*) bidireccionales. Se basa en la idea de que la atención se traspasa del contexto a la pregunta (Context2Query) y de la pregunta al contexto (Query2Context), con la obtención de una matriz de similitud con distintas funciones entre palabras del contexto y la pregunta (producto escalar, combinación lineal, bilineal, y con perceptrón). Después, para la capa de modelado se obtiene una matriz de similitud con biLSTMs (y así la suma ponderada de las palabras más importantes del contexto con respecto a la pregunta) que se traspasa a la capa de predicción de la respuesta: una capa densa con *softmax* y LSTM+softmax de la palabra de comienzo y fin de la respuesta, respectivamente.

En los anexos (Seo et al., 2016) se hace hincapié en los errores cometidos sobre *SQuAD* y *CNN Daily Mail* datasets, con un análisis por tipo de error, ratio y ejemplo. Ordenados por ratio de forma descendente (de 50 % a 2 %), los errores ejemplificados son: imprecisión en el comienzo y final de la respuesta, error por estructura sintáctica complicada y ambigüedades, problema por parafrases, necesidad de conocimiento externo para responder, presencia de la respuesta en múltiples oraciones y por preprocesado incorrecto.

2.3.3 Modelos Neuronales de Lenguaje Pre-Entrenados

El uso de los Modelos Neuronales de Lenguaje Pre-Entrenados o Modelos Generales de Lenguaje está motivado por el planteamiento de *transfer learning* impulsado por el área de Visión Artificial y Deep Learning (Ruder, 2018a). La transferencia de conocimiento se realiza a partir de una arquitectura y las características de representación (parámetros numéricos) obtenidos en una red profunda entrenada con un conjunto de datos masivo para inicializar un nuevo modelo (Ruder, 2017). Tras la adición y refinamiento de las últimas capas del nuevo modelo (*fine-tunning* (Ruder et al., 2019)) con el conjunto de datos específico (de menor tamaño), se obtiene una segunda versión del modelo eficiente sin necesidad de realizar un re-entrenamiento completo y con menor tiempo de entrenamiento, ya que solo es necesario para el ajuste final de los parámetros.

Las tareas resueltas con Modelos Neuronales de Lenguaje sigue en aumento y constante evolución (Ruder et al., 2019). Desde la traducción automática, la clasificación de secuencias de textos para análisis sintáctico y reconocimiento de entidades, generación de textos, representación avanzada para clasificación, para los sistemas de Pregunta-Respuesta y Generación de Lenguaje Natural (Youngy et al., 2018).

Para ello ha sido clave la publicación de algoritmos, modelos y arquitecturas para representación de textos (*embedding*), en diversas etapas (Ruder, 2018b). Los modelos neuronales de lenguaje se originan en 2001 con el reto

de representar el lenguaje a partir del objetivo de predecir la siguiente palabra a partir de un contexto de palabras con un perceptrón multicapa (*feed forward network*, ver 2.6), mejorando las técnicas de n-gramas con modelos probabilísticos HMM (Jurafsky and Martin, 2009-2021).

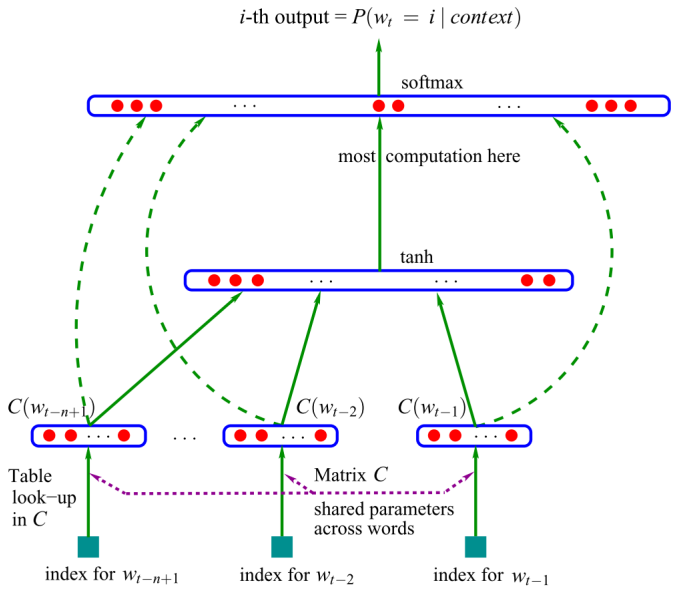


Figura 2.6: *Origen de los modelos neuronales de lenguaje: feed forward newtork (Bengio et al., 2003)*

Posteriormente, a partir del año 2008 se proponen modelos que puedan emplearse para múltiples tareas en análisis sintáctico, detección de entidades y clasificación con componentes convolucionales y secuenciales (CNN, RNN, LSTM, biLSTM y más en el estudio Collobert et al. (2011)), y arquitecturas más complejas como *seq2seq* y modelos de atención (Vaswani et al., 2017) que emplean en su primera capa los modelos pre-entrenados de *word embedding*, obteniendo mejores resultados en las tareas de traducción, análisis sintáctico,

reconocimiento de entidades, análisis de sentimiento, clasificación de textos (Youngy et al., 2018). Los *word embedding* más utilizados en los Modelos Neuronales de Lenguaje se publican a partir del año 2013: *word2vec* (Google), *GloVe* (Stanford), *FastText* (Facebook) y *ElMo* (Peters et al., 2018).

A partir del año 2018 arranca el uso Modelos Neuronales de Lenguaje pre-entrenados con recursos de la web masivos no anotados y se hacen disponibles al público en general por diversas organizaciones como OpenAI con su modelo *GPT* (Radford et al., 2018) y Google en el caso del modelo *BERT* (Devlin et al., 2018), y más modelos basados en *Transformer*, que también superan resultados previos en Pregunta-Respuesta.

Transformer es un componente de atención publicado en el estudio (Vaswani et al., 2017) que permite mejora en la representación jerárquica y priorizada de la secuencia de palabras presentes a nivel de oraciones. Consta a su vez de dos bloques enlazados (*Encoder* y *Decoder*) y emplea técnicas de atención con varios canales (*multi-head attention*) capaces de representar y extraer características de forma focalizada y contextualizada en distintas partes de la secuencia de entrada, de forma que la representación de las palabras es abstraída en distintos niveles gracias a su profundidad de capas (ver figura 2.7).

Este avance ha sido posible gracias a mayor capacidad de cómputo a nivel de infraestructuras y al ecosistema tecnológico disponible en Python. Desde la publicación de *Tensorflow* (2015, Google) y *PyTorch* (2016, Face-

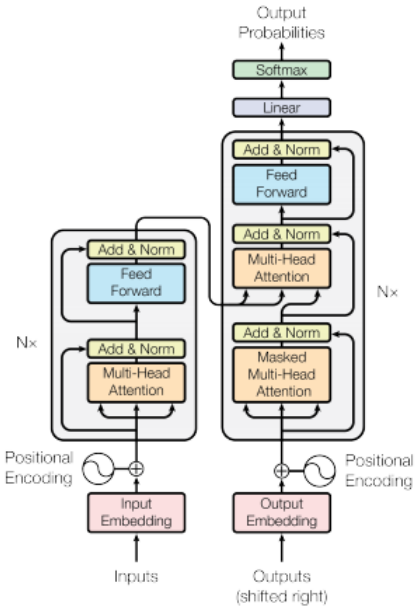


Figura 2.7: *Arquitectura de Transformer (Vaswani et al., 2017)*

book) hay disponibles librerías a alto nivel para facilitar el entrenamiento y prototipado rápido de arquitecturas para modelos neuronales (*Keras*, *spaCy* (Honnibal and Montani, 2016-2021), *Stanza* (Qi et al., 2020)) y con los modelos neuronales pre-entrenados públicos (*HuggingFace* Wolf et al. (2019)).

La librería abierta de HuggingFace de Wolf et al. (2019) dispone de Modelos Neuronales de Lenguaje de múltiples publicaciones y organizaciones con pre-entrenamiento y refinado (*fine-tunning*) en diversos números de parámetros (*base*, *large*) como *BERT*, *RoBERTa* y *T5* nuevas variaciones de los mismos, también los modelos para generación de lenguaje *XLNet* y *GPT*. Todo ello se hace disponible para su uso en la librería mediante la creación de *pipelines* abstractas para cada tarea de lenguaje, en este caso *question-*

answering, que devuelve la respuesta en forma de *offsets* de comienzo y fin del contexto de entrada, respuesta y confianza de predicción.

Con ello, se ha originado una nueva rama de estudio denominada como *Bertology* (Wang and Zhang, 2019-2021) donde se originan múltiples modelos: *RoBERTa*, *XLNet*, *T5*, *AlBERT*, *DistillBERT*, *LUKE* y más como se puede observar en la figura 2.8. Además, se motiva la evaluación de estos modelos en múltiples tareas de Procesamiento de Lenguaje Natural gracias a los *benchmark* como *GLUE* (Wang et al., 2018) que surgen para analizar la capacidad de entendimiento de lenguaje natural (*NLU*) con tareas como el soporte a paráfrasis y similitud semántica textual, fenómenos lingüísticos frecuentes también en Pregunta-Respuesta.

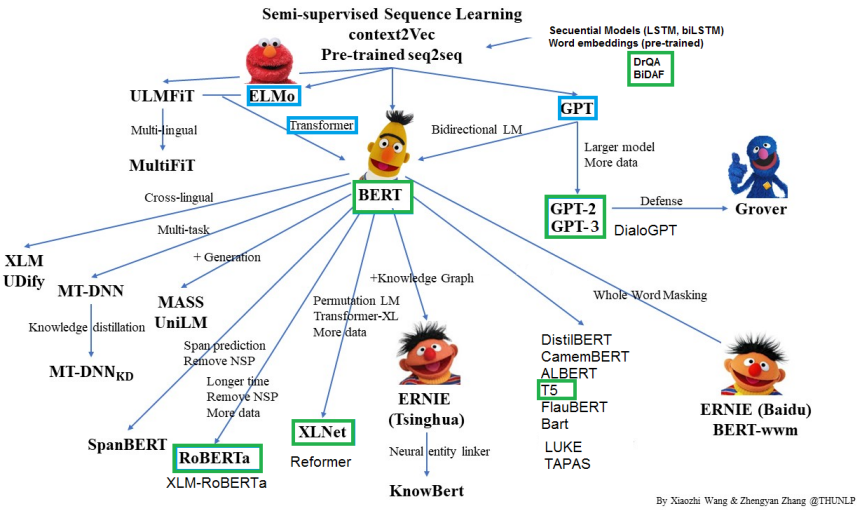


Figura 2.8: *Modelos de la rama de estudio de Bertology (Wang and Zhang, 2019-2021)*

A continuación se profundiza en los Modelos Neuronales de Lenguaje

pre-entrenados más evaluados en Búsqueda de Respuestas y Comprensión Lectora.

2.3.3.1 GPT

OpenAI propone en el trabajo Radford et al. (2018) el planteamiento de entrenamiento semi-supervisado basado en pre-entrenamiento. Este novedoso cambio de perspectiva evita tener que generar colecciones de gran tamaño con gran esfuerzo para su anotación manual. Se deja atrás el planteamiento de entrenamientos de los modelos de un dominio específico para experimentar qué tipo de transferencia de características y de conocimiento general se obtiene gracias a la representación general del lenguaje. Nuevamente, el planteamiento de aprendizaje con pre-entrenamiento permite mejorar en eficiencia, como ya se demostró en estudios previos de representación de lenguaje a nivel de palabra, *word embedding* (Peters et al., 2018).

En primer lugar se realiza el entrenamiento del modelo de forma no supervisada empleando un corpora masivo no anotado, *BookCorpus*, para después refinarlo en la tarea concreta con un corpus de menor tamaño de forma supervisada. En el caso de la tarea de Pregunta-Respuesta, se emplea la colección *RACE* (Lai et al., 2017) y para similitud semántica textual la colección *QQP* perteneciente al benchmark de *GLUE* (Wang et al., 2018).

En vez de inicializar el modelo con un *word embedding*, se emplea el algoritmo *Byte Pair Encoding*, para representar características morfológicas

frecuentes a nivel de carácter y superar las limitaciones de representación ante palabras desconocidas (*out of vocabulary*, OOV).

Basado en *Transformer*, el modelo *Generative Pre-trained Transformer* (GPT) permite extraer características con el componente *Decoder* (ver figura 2.9) a partir de secuencias de lenguaje vistas en las narraciones y libros de aventuras, ficción, románticos y más géneros de *BookCorpus*, diferenciándose así del coetáneo Modelo Neuronal de Lenguaje *ELMo* que emplea un conjunto de entrenamiento basado en oraciones analizadas de forma bidireccional, *1 Billion Word Benchmark* (Peters et al., 2018). Ambos modelos permiten mayor contextualización de conocimiento, superando las limitaciones de los *word embedding* ante polisemia y cambios de sentido de las palabras por el contexto.

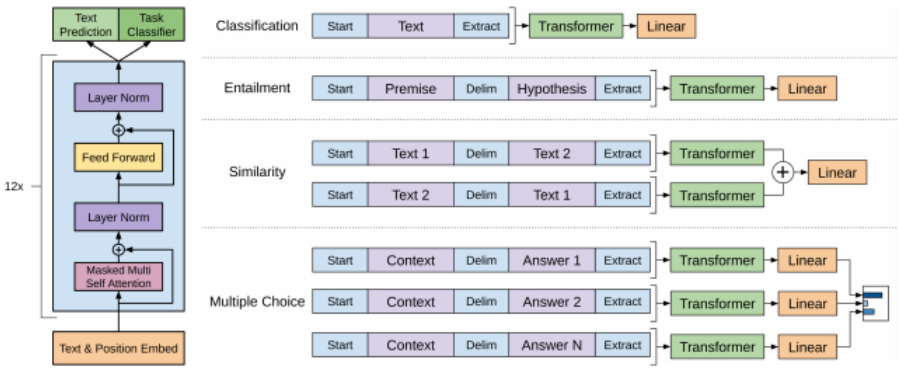


Figura 2.9: *Arquitectura de GPT-1 y entradas para múltiples tareas de Procesamiento de Lenguaje (Radford et al., 2018)*

En su segunda versión, *GPT-2*, el modelo alcanza 1.5 billones de parámetros y es pre-entrenado con corpus de 8 millones de páginas web (denominado

WebText), sin incluir entradas de Wikipedia. Este modelo afronta la tarea de Generación de Lenguje Natural, permitiendo redactar artículos en lenguaje natural con estructuras sintácticas coherentes a partir de una oración semilla, motivo por el que los autores no lo hicieron disponible a la comunidad inmediatamente para evitar un mal uso del mismo.

En el estudio Radford et al. (2019), GPT-2 se evalúa para la tarea de Pregunta-Respuesta con la colección *CoQA*, para la que se obtiene un rendimiento similar a la eficiencia humana. También se compara la confianza obtenida en su respuesta para la colección de *Natural Questions* (preguntas del buscador de Google sobre artículos de Wikipedia), con una exactitud del 63.1 % sobre el 1 % de las preguntas con mayor confianza, ejemplificadas en la figura 2.10. La distribución de confianza es balanceada a pesar de que las respuestas de dicha colección están contenidas en Wikipedia, entradas que el modelo no dispone en su pre-entrenamiento. Así, se propone una forma de evaluar la trasferencia de conocimiento general que recibe el modelo en su pre-entrenamiento a partir de la realización de preguntas factuales al mismo y observando la confianza obtenida en las respuestas más allá de su corrección o de métricas de solapamiento de la respuesta correcta y la generada (Radford et al., 2019).

En su tercera versión, OpenAI se propone evitar la fase de refinamiento supervisado (*fine-tunning*) proponiendo un modelo de mayor número de parámetros para que con pocos o un ejemplo que ilustre la tarea a realizar el modelo pueda dar respuesta (*few-shot* y *one-shot*). Para ello, GPT-3 se

Question	Generated Answer	Correct	Probability
Who wrote the book the origin of species?	Charles Darwin	✓	83.4%
Who is the founder of the ubuntu project?	Mark Shuttleworth	✓	82.0%
Who is the quarterback for the green bay packers?	Aaron Rodgers	✓	81.1%
Panda is a national animal of which country?	China	✓	76.8%
Who came up with the theory of relativity?	Albert Einstein	✓	76.4%
When was the first star wars film released?	1977	✓	71.4%
What is the most common blood type in sweden?	A	✗	70.6%
Who is regarded as the founder of psychoanalysis?	Sigmund Freud	✓	69.3%
Who took the first steps on the moon in 1969?	Neil Armstrong	✓	66.8%
Who is the largest supermarket chain in the uk?	Tesco	✓	65.3%
What is the meaning of shalom in english?	peace	✓	64.0%
Who was the author of the art of war?	Sun Tzu	✓	59.6%
Largest state in the us by land mass?	California	✗	59.2%
Green algae is an example of which type of reproduction?	parthenogenesis	✗	56.5%
Vikram samvat calender is official in which country?	India	✓	55.6%
Who is mostly responsible for writing the declaration of independence?	Thomas Jefferson	✓	53.3%
What us state forms the western boundary of montana?	Montana	✗	52.3%
Who plays ser davos in game of thrones?	Peter Dinklage	✗	52.1%
Who appoints the chair of the federal reserve system?	Janet Yellen	✗	51.5%
State the process that divides one nucleus into two genetically identical nuclei?	mitosis	✓	50.7%
Who won the most mvp awards in the nba?	Michael Jordan	✗	50.2%
What river is associated with the city of rome?	the Tiber	✓	48.6%
Who is the first president to be impeached?	Andrew Johnson	✓	48.3%
Who is the head of the department of homeland security 2017?	John Kelly	✓	47.0%
What is the name given to the common currency to the european union?	Euro	✓	46.8%
What was the emperor name in star wars?	Palpatine	✓	46.5%
Do you have to have a gun permit to shoot at a range?	No	✓	46.4%
Who proposed evolution in 1859 as the basis of biological development?	Charles Darwin	✓	45.7%
Nuclear power plant that blew up in russia?	Chernobyl	✓	45.7%
Who played john connor in the original terminator?	Arnold Schwarzenegger	✗	45.2%

Figura 2.10: *Preguntas y respuestas generadas con GPT-2 con mayor confianza (Radford et al., 2019)*

pre-entrena con más dominios obtenidos de la web con el corpora *Common Crawl* (45 TB) tras mejorar su calidad con técnicas de filtrado y retirado de duplicados a nivel de documento (reduciendo a 570 GB), además de los corpus anteriores.

La arquitectura y modelo, basado en la de GPT-2, se amplía en el número de capas y parámetros obtenidos en su *Byte Pair Encoding* (BPE) obteniendo el mayor modelo con 175 billones de parámetros. Con ello, además de mejorar la calidad de los artículos generados, también se amplía el número de tareas en Generación de Lenguaje Natural que este modelo realiza: generación de código a partir de una descripción de lenguaje natural, resumen

de textos, traducción de lenguaje natural a comandos válidos para bases de datos (SQL). Sin embargo, no se obtienen resultados eficientes en tareas de razonamiento e inferencia de lenguaje natural ni para Comprensión Lectora, como muestra su evaluación sobre la colección *RACE* (Brown et al., 2020).

2.3.3.2 BERT

El modelo de lenguaje pre-entrenado de Google AI, *Bidirectional Encoder Representations from Transformers* (*BERT*) (Devlin et al., 2018), supera en eficiencia a los modelos previos en distintas tareas y conjuntos de datos asociados de Procesamiento de Lenguaje Natural además de sistemas de Pregunta Respuesta. Se proponen dos versiones de *BERT*: *base* de 110 millones de parámetros y *large* de 340 millones de parámetros en total.

Su arquitectura consiste en dos *Transformer* bidireccionales para la representación a nivel de oraciones, a diferencia del modelo GPT que solo emplea unidireccionalidad y un solo componente. Para su inicialización se utilizan los embeddings obtenidos con un algoritmo similar a *Byte Pair Encoding*, *WordPiece*, sobre el conjunto *BookCorpus* y Wikipedia, con un total de 30.000 tokens de entrada.

Sobre el conjunto de datos para su pre-entrenamiento, se modela el lenguaje al aplicar una máscara para predecir cierta palabra a partir del contexto, y como segunda tarea se predice la siguiente frase a partir de la primera, por lo que es necesario tener como entrada un *embedding* por segmento y

posición para cada par de oraciones, ver figura 2.11. A partir del modelo pre-entrenado se refinan los pesos para la tarea particular superando eficiencias obtenidas con el entrenamiento desde el comienzo de componentes secuenciales u otras arquitecturas.

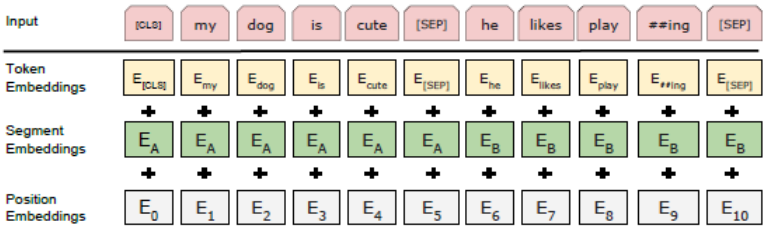


Figura 2.11: *Embeddings empleados en BERT (Devlin et al., 2018)*

Se ha evaluado sobre el benchmark *GLUE* del estudio Wang et al. (2018) (ver la figura 2.15), y para la tarea de Comprensión Lectora de Pregunta-Respuesta, con la colección de *SQuAD* (Rajpurkar et al., 2016) (ver la figura 2.13).

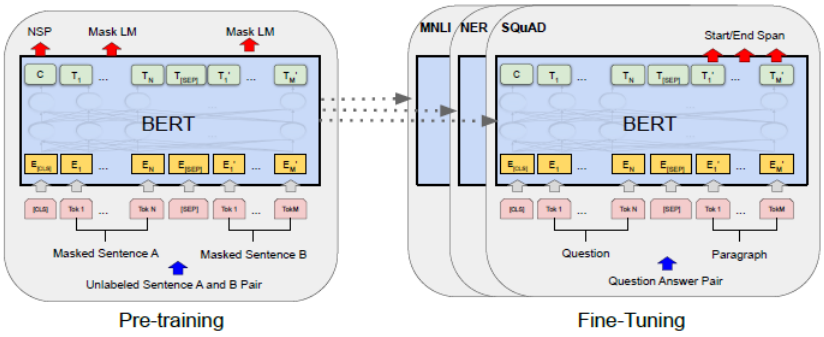


Figura 2.12: *Procesos de pre-entrenamiento y refinamiento de BERT (Devlin et al., 2018)*

A raíz de este modelo se fundamenta una línea de trabajo y de investigación *post-BERT* con pre-entrenamientos con más corpora, variaciones y optimizaciones de los planteamientos, como *RoBERTa* y *XLNet*.

2.3.3.3 RoBERTa

En el estudio *A Robustly Optimized BERT Pretraining Approach* (Liu et al., 2019) se replica el pre-entrenamiento de *BERT* ya que es posible entrenarlo mayor tiempo, con mayor tamaño de *batches*, con más corpora (se añade *Common Crawl News*, *Open Web Text* y *Stories*), se simplifica el objetivo de la predicción de la siguiente oración, se mejora el soporte a secuencias largas (ver eficiencias en 2.13) y se aplica una máscara dinámica sobre el conjunto de entrenamiento.

Model	**data**	**bsz**	**steps**	**SQuAD** (v1.1/2.0)
RoBERTa				
with BOOKS + WIKI	16GB	8K	100K	93.6/87.3
+ additional data (§3.2)	160GB	8K	100K	94.0/87.7
+ pretrain longer	160GB	8K	300K	94.4/88.7
+ pretrain even longer	160GB	8K	500K	**94.6/89.4**
$\text{BERT}_{\text{LARGE}}$				
with BOOKS + WIKI	13GB	256	1M	90.9/81.8
$\text{XLNet}_{\text{LARGE}}$				
with BOOKS + WIKI	13GB	256	1M	94.0/87.8
+ additional data	126GB	2K	500K	94.5/88.8

Figura 2.13: *Comparación entre RoBERTa, BERT, XLNet y su evaluación sobre SQuAD (Liu et al., 2019)*

Todas estas variaciones permiten superar las métricas de eficiencia obtenidas con los modelos de *BERT* y similares a *XLNet* en su evaluación sobre

los conjuntos del benchmark de *GLUE* (ver figura 2.15) (Wang et al., 2018), y *SQuAD*, (Rajpurkar et al., 2016) (Rajpurkar et al., 2018).

2.3.3.4 XLNet

En el estudio de *BERT* se asume que las partículas (*token*) predichos con la contextualización bidireccional son independientes entre sí por lo que no se tiene en cuenta la gran dependencia de orden y secuencia de las palabras, condiciones intrínsecas de la estructura del lenguaje natural.

El estudio Yang et al. (2019) propone un método de pre-entrenamiento autorregresivo generalizado que permite al modelo aprender a partir de los contextos bidireccionales maximizando la probabilidad esperada en todas las permutaciones en su factorización. Gracias a la operación de permutación, el contexto de cada posición puede depender tanto de las palabras de la izquierda como de la derecha, se aprende a utilizar la información contextual de todas las posiciones para capturar el contexto bidireccional, también ante secuencias largas de lenguaje.

Como modelo lingüístico autorregresivo, no se depende de sesgos o corrupciones en las máscaras aplicadas o de los ejemplos del propio corpus, se elimina la suposición de independencia previa gracias a que el objetivo autorregresivo permite emplear la regla del producto para factorizar la probabilidad conjunta de los *token* predichos, ejemplificado en la figura 2.14.

XLNet supera a BERT en múltiples tareas incluyendo la respuesta a pre-

$$\mathcal{J}_{\text{BERT}} = \log p(\text{New} \mid \text{is a city}) + \log p(\text{York} \mid \text{is a city}),$$
$$\mathcal{J}_{\text{XLNet}} = \log p(\text{New} \mid \text{is a city}) + \log p(\text{York} \mid \text{New}, \text{is a city}).$$

Figura 2.14: *Ejemplo de factorización de BERT y de XLNet (Yang et al., 2019)*

guntas, la inferencia del lenguaje natural, el análisis de sentimientos, gracias a su evaluación realizada con los conjuntos del benchmark de *GLUE* del estudio (Wang et al., 2018) (figura 2.15), y *SQuAD* (Rajpurkar et al., 2018) (figura 2.16).

Model	MNLI	QNLI	QQP	RTE	SST-2	MRPC	CoLA	STS-B
Single-task single models on dev								
BERT [2]	86.6/-	92.3	91.3	70.4	93.2	88.0	60.6	90.0
RoBERTa [21]	90.2/90.2	94.7	92.2	**86.6**	96.4	**90.9**	68.0	92.4
XLNet	**90.8/90.8**	**94.9**	**92.3**	85.9	**97.0**	90.8	**69.0**	**92.5**
Multi-task ensembles on test (from leaderboard as of Oct 28, 2019)								
MT-DNN* [20]	87.9/87.4	96.0	89.9	86.3	96.5	92.7	68.4	91.1
RoBERTa* [21]	90.8/90.2	98.9	90.2	88.2	96.7	92.3	67.8	92.2
XLNet*	**90.9/90.9**†	**99.0**†	**90.4**†	**88.5**	**97.1**†	**92.9**	**70.2**	**93.0**

Figura 2.15: *Comparación entre RoBERTa, BERT, XLNet y su evaluación GLUE (Yang et al., 2019)*

SQuAD2.0	EM	F1	SQuAD1.1	EM	F1
Dev set results (single model)					
BERT [10]	78.98	81.77	BERT† [10]	84.1	90.9
RoBERTa [21]	86.5	89.4	RoBERTa [21]	88.9	94.6
XLNet	**87.9**	**90.6**	XLNet	**89.7**	**95.1**
Test set results on leaderboard (single model, as of Dec 14, 2019)					
BERT [10]	80.005	83.061	BERT [10]	85.083	91.835
RoBERTa [21]	86.820	89.795	BERT* [10]	87.433	93.294
XLNet	**87.926**	**90.689**	XLNet	**89.898**‡	**95.080**‡

Figura 2.16: *Evaluación con SQuAD para BERT, XLNet y RoBERTa (Yang et al., 2019)*

2.3.3.5 T5

La idea diferencial del estudio *Text-to-Text Transfer Transformer* (*T5*) (Raffel et al., 2019) es tratar cada problema de Procesamiento de Lenguaje Natural (pregunta-respuesta, modelado de lenguaje, clasificación textual, resumen automático, análisis de sentimiento, traducción y similitud semántica textual, etc.) con un texto de entrada y producir un nuevo texto como salida con un único *framework* (ver figura 2.17).

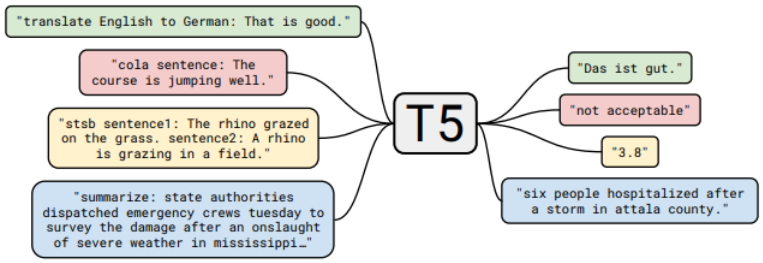


Figura 2.17: *Planteamiento de Text-to-Text Transfer Transformer, T5 (Raffel et al., 2019)*

Para ello, se compara el planteamiento del uso de un único modelo para todas las tareas (*multi-task learning*) versus al planteamiento de un modelo formado por el conjunto de varios modelos (*ensemble learning*) preentrenados y con *fine-tunning* en cada tarea, obteniendo mejor resultado con la segunda opción. Además, en los experimientos realizados se demuestra que la combinación de modelos ajustados a partir de mismo modelo preentrenado de base tuvo peores resultados que el preentrenamiento y el ajuste fino de todos los modelos por separado.

El estudio evalúa la escalabilidad y limitaciones de los Modelos Neuronales de Lenguaje con perspectiva de *deep learning*: arquitecturas, estrategias de máscaras, dropout, número de parámetros empleados, número de épocas de entrenamiento y tiempo empleado, entre otros criterios (Raffel et al., 2019). Además, confirma que el pre-entrenamiento con datos no etiquetados del dominio mejora el rendimiento en tareas posteriores, como se demuestra en la mejora en eficiencia del pre-entrenamiento con Wikipedia en comparación con otros datasets y la evaluación posterior con *SQuAD*, basado en entradas de Wikipedia (Rajpurkar et al., 2016).

T5 se pre-entrena con el corpus *Colossal Clean Crawled Corpus (C4)* obtenido a partir de 20TB de datos de la web, reducidos a 750GB tras su filtrado y limpiado. Se confirma que las métricas obtenidas en la evaluación con colecciones como *SQuAD* y el benchmark *GLUE* descienden con la reducción del número de *tokens* empleados en el pre-entrenamiento; se realizan experimentos de pre-entrenamiento con distinto número de *tokens* de entrada contemplados en el corpus, desde 2^{23} hasta el total de 2^{35}. Estos experimentos se realizan sobre un número reducido de *tokens* de entrada en comparación con *BERT* (137B *tokens*) y *RoBERTa* (2.2T) (Raffel et al., 2019).

Sobre el modelo base de T5, se amplían el número de parámetros considerando mayor número de parámetros de embedding, mayor número de capas de atención y capas para fine-tunning consiguiendo así mayor eficiencia en cuanto a métricas, ver la tabla 2.9. Así, el modelo T5-Base cuenta con 220

millones de parámetros, la versión T5-Small 60 millones de parámetros, la versión Large 770 millones de parámetros.

Model	GLUE Average	STS-B Pearson	QQP F1	QQP Accuracy	SQuAD EM	SQuAD F1
Previous best	89.4[a]	92.7[b]	74.8[c]	**90.7**[b]	90.1[a]	95.5[a]
T5-Small	77.4	85.6	70.0	88.0	79.10	87.24
T5-Base	82.7	89.4	72.6	89.4	85.44	92.08
T5-Large	86.4	89.9	73.9	89.9	86.66	93.79
T5-3B	88.5	90.6	74.4	89.7	88.53	94.95
T5-11B	**90.3**	**93.1**	**75.1**	90.6	**91.26**	**96.22**

Tabla 2.9: *Evaluación de Text-to-Text Transfer Transformer, T5, en GLUE y SQuAD (Raffel et al., 2019). Eficiencias con modelos basados en BERT (a) ALBERT, (b) StructBERT, (c) XLNet.*

Se obtiene como conclusión que el entrenamiento de un modelo con menor número de parámetros con un corpus de gran tamaño es superado por el entrenamiento de un modelo con mayor número de parámetros con menos épocas y menos tiempo de entrenamiento (Raffel et al., 2019).

2.4 Análisis de dificultad preliminar

Los sistemas de Pregunta-Respuesta se enfrentan a la capacidad de analizar y entender el lenguaje, resolver la necesidad de conocimiento previo (de ámbito general o dominio específico), analizar relaciones y pasajes de múltiples líneas o documentos. En su fase final, al dar una respuesta, proporcionar el tipo y forma correcta a partir de la selección de respuesta entre múltiples opciones, no dar respuesta (ante preguntas no contestables) o su extracción del texto.

Los primeros métodos de evaluación en Búsqueda de Respuestas proponen exámenes de comprensión lectora o test de lectura sobre los que considerar si un modelo tiene la habilidad de responder a variedad de preguntas de forma precisa y escueta ante distintos dominios (noticias, textos científicos), niveles de complejidad, razonamientos y necesidad de conocimiento general (Clark and Etzioni, 2016). Un ejemplo de anotación por dificultad se puede ver en la colección *RACE* del estudio Lai et al. (2017) ya que proviene por su creación de exámenes para cursos académicos (*high* y *medium*).

En general, los estudios analizan errores cometidos por los modelos de Pregunta-Respuesta para la colección propuesta mediante la inspecciona una muestra reducida, agrupando preguntas por razonamientos que suponen dificultad (ver sección 2.2).

Evaluar si un modelo de Pregunta-Respuesta será capaz de responder a gran variedad de preguntas, de diversas complejidades, demostrando capacidad de sentido común y conocimiento general y generalizar a nuevo conocimiento o dominio que no ha recibido en su entrenamiento es la motivación para crear múltiples colecciones de Pregunta-Respuesta y evaluar las dificultades que suponen para el modelo de una forma estandarizada. Por ello, se propone su análisis y comparativa con un análisis preliminar de dificultad además de detallar las métricas de evaluación obtenidas con los Modelos Neuronales de Lenguaje de Búsquedas de Respuestas.

2.4.1 Análisis y comparativa de las colecciones

Todas las colecciones contempladas en este trabajo se han creado con recursos en inglés, y en su mayoría gracias a crowd-sourcing para conseguir mayor cantidad de ejemplos, superando así la limitación principal de tamaño reducido de las primeras colecciones como *MCTest*.

Sin embargo, esta metodología de creación da lugar a colecciones con planteamiento de pregunta-respuesta extractivo en un único dominio, como *CNN Daily Mail* y *SQuAD* (ver 2.2.3), entre otras. Las principales asunciones implícitas de este planteamiento es la escasa variedad lingüística entre la pregunta y la respuesta ya que al crear las preguntas los usuarios tienen tendencia a emplear términos similares a los presentados en la respuesta; no se permite evaluar los modelos sobre múltiples dominios y conocimiento general o específico no recopilado en el entrenamiento de los mismos, ya que por su definición, requieren que la respuesta esté explícitamente recogida en el texto a evaluar. Esto facilita que los modelos contesten gracias a coincidencia léxica sin un razonamiento profundo sobre múltiples textos o documentos (Rogers et al., 2020), inferencia ni necesidad de conocimiento general (Clark and Etzioni, 2016).

Para superar estas limitaciones, las colecciones más recientes incluyen nuevas metodologías de creación y retos como la resolución de pregunta-respuesta sobre conversaciones y múltiples dominios (*CoQA*, *QuAIL*) y el reto de entendimiento de preguntas similares de forma masiva (*Quora Ques-*

tion Pairs).

Como resumen, los cuadros 2.10 y 2.11 recopilan las características de las colecciones contempladas en este trabajo con perspectiva cualitativa y cuantitativa respectivamente.

Colección	Preguntas	Tipo QA	Origen
MCTest	Factuales	Múltiple	Historias Compr. Lect.
CNN Daily Mail	Factuales	Extractivo	Artículos
SQuAD	Abierto	Extractivo, Sin Resp.	Wikipedia
RACE	Abierto	Múltiple	Exámenes Compr. Lect.
NewsQA	Abierto	Extractivo, Sin Resp.	CNN
QQP	Abierto	Similitud	Quora, GLUE
CoQA	Abierto	Extractivo	Múltiples dominios
Natural Questions	Abierto	Recuperación	Wikipedia
QuAIL	Abierto	Múltiple	Múltiples dominios

Tabla 2.10: *Comparativa cualitativa de las colecciones de Pregunta-Respuesta*

Respecto al análisis estadístico de las colecciones, el estudio Lai et al. (2017) recopila información de razonamientos y aspectos lingüísticos contemplados manualmente en 1.000 ejemplos de las colecciones *RACE*, *CNN Daily Mail* y *SQuAD 1.1*, informados en la tabla 2.12.

Sin embargo, estos estudios no entran en la caracterización en detalle del lenguaje de la colección por su formulación ni en dificultades por más tipos de fenómenos lingüísticos de forma generalizada.

Colección	Textos	Preguntas	train	dev	test
MCTest	660	2.640	-	-	-
CNN Daily Mail	90.266/197k	380k/879k	-	-	-
SQuAD 2.0	536	142.192	130.319	11.873	-
RACE	27.933	97.687	87.866	4.887	4.934
NewsQA	90.266	119.633	107.674	5.988	5.971
QQP	-	404.301	-	-	-
CoQA	8k	127k	-	-	-
Natural Questions	-	315.203	-	-	-
QuAIL	800	15k	-	-	-

Tabla 2.11: *Comparativa cuantitativa de las colecciones de Pregunta-Respuesta*

Dataset	RACE-M	RACE-H	RACE	CNN	SQUAD
Word Matching	29.4%	11.3%	15.8%	13.0%†	39.8%*
Paraphrasing	14.8%	20.6%	19.2%	41.0%†	34.3%*
Single-Sentence Reasoning	31.3%	34.1%	33.4%	19.0%†	8.6%*
Multi-Sentence Reasoning	22.6%	26.9%	25.8%	2.0%†	11.9%*
Ambiguous/Insufficient	1.8%	7.1%	5.8%	25.0%†	5.4%*

Tabla 2.12: *Información estadística de razonamientos y aspectos lingüísticos en las colecciones RACE, CNN Daily Mail (Chen et al., 2016) en comparación con SQuAD (Lai et al., 2017)*

2.4.2 Dificultades de los Modelos Neuronales de Lenguaje

Las dificultades observadas en el estudio del estado del arte se obtienen a partir de la comparación de métricas de evaluación de Aprendizaje Automático (como exactitud o F1) obtenidas en los modelos. No se evalúa la dificultad de cada colección empleada en detalle, ya que estas métricas se

limitan a contabilizar preguntas respondidas correctamente.

Gracias al estudio de Reddy et al. (2019) se observa que el modelo DrQA tiene dificultades ante las respuestas del tipo sí y no, en menor medida para determinar si la pregunta es contestable o no (supera el 50 % de F1). BiDAF y DrQA superan los modelos secuenciales basados en LSTM como *seq2seq* para localizar respuestas que contienen entidades nombradas, frases nominales, respuestas del tipo numéricas o de fechas (Manning, 2020).

A su vez, los Modelos Generales de Lenguaje pre-entrenados superan en eficiencia a los Modelos Neuronales de Lenguaje entrenados para tareas y dominios específicos basados en convoluciones y modelos secuenciales, a modelos inicializados con representaciones a nivel de palabra (*word embedding* como *word2vec*, *GloVe*, o *ElMo*), con representaciones a nivel morfológico, como *FastText*, o enriquecidos con más información de características lingüísticas con preprocesados de lenguaje (*embeddings* por posición, etiquetas sintácticas, reconocimiento de entidades nombradas, etc.); ya que además de detectar relaciones sintácticas y capturar relaciones semánticas dependientes del contexto, superan las dificultades previas en entendimiento de lenguaje y de representación de conocimiento general (Peters et al., 2018). Muestran mejor rendimiento ante paráfrasis, pragmática y son capaces de resolver la necesidad de conocimiento general gracias a su pre-entrenamiento, incluso ante su evaluación en múltiples dominios, como muestra el estudio Rogers et al. (2020), siendo ligeramente inferior su resultado solamente ante la detección de preguntas no contestables, solo superados por estrategias de similitud

semántica textual (*AvgCos*).

Sin embargo, todos los modelos son sensibles a variaciones adversas y presentan errores ante razonamientos lógicos o inferencias, como muestran las pruebas de estrés con la modificación ruidosa de *SQuAD* a nivel de fragmentos (*AddOneSent*, *AddSent*, *AddAny*) (Aspillaga et al., 2020) ya que la exactitud disminuye entre un 18.1 % y 32.2 % para los modelos *BERT*, *XLNet* y *RoBERTa* (modelos basados en *Transformer*).

Además, la exactitud se reduce drásticamente ante errores ortográficos y errores de preprocesado, entre un 46.3 % y un 72 % en los modelos basados en *Transformer*, y más ante reemplazamientos y variaciones aleatorias a nivel de palabra, entre un 93.3 % y 96.2 %, como se observa en la tabla 2.13. Se considera que su exactitud se reduce en menor medida ante ruidos de una única permutación de caracteres de la palabra ya que los modelos soportan ligeramente los errores ortográficos de este tipo por ser los errores naturales presentes en las colecciones (Aspillaga et al., 2020).

Model	Original Dev	Concatenative Adversaries				Noise Adversaries				
		AddOne-Sent	AddSent	AddAny	Add-Common	Swap	Middle Random	Fully Random	Keyboa-rd Typo	Natural
RoBERTa	**85.8**	**70.3** [18.1]	61.5 [28.3]	77.3 [9.9]	**84.3** [1.7]	**46.1** [46.3]	**32.2** [62.5]	3.3 [96.2]	**30.4** [64.6]	54.9 [36.0]
XLNet	85.2	67.7 [20.5]	**61.6** [27.7]	**78.8** [7.5]	83.0 [2.6]	43.0 [49.5]	31.9 [62.6]	4.4 [94.8]	27.2 [68.1]	**57.4** [32.6]
BERT	82.5	64.6 [21.7]	55.9 [32.2]	71.4 [13.5]	81.1 [1.7]	33.8 [59.0]	28.6 [65.3]	**5.5** [93.3]	23.1 [72.0]	47.7 [42.2]
Match-LSTM	60.8	30.0 [50.7]	24.8 [59.2]	35.7 [41.3]	52.5 [13.7]	17.8 [70.7]	20.2 [66.8]	4.1 [93.3]	9.4 [84.5]	19.7 [67.6]

Tabla 2.13: *Exactitud obtenida con modelos basados en Transformer sobre SQuAD v1.1 de test (dev), Aspillaga et al. (2020).*

Son escasos los estudios en los que se realice una caracterización de la colección a partir de la anotación con técnicas de Procesamiento de Lenguaje Natural de todos los ejemplos de pregunta-respuesta, el análisis estadístico de resultados para una posterior comparación de fenómenos de lenguaje que presentan las colecciones y dificultades que presentan para los modelos.

2.5 Métodos de Procesamiento de Lenguaje Natural

El área de Procesamiento de Lenguaje Natural permite realizar el análisis a gran escala de fuentes textuales digitales, desde detectar el idioma, traducir, obtener palabras frecuentes y categorías, analizar el sentimiento y reconocer acontecimientos y entidades hasta caracterizar el lenguaje por su formulación a partir del análisis léxico, sintáctico y semántico.

Con técnicas estadísticas y de Aprendizaje Automático para representación textual, es posible clasificar textos para un objetivo de forma supervisada o agruparlos por su similitud de forma no supervisada, obtener temáticas y generar resúmenes. Como se ha observado en estudios recientes de Modelos Neuronales de Lenguaje estos modelos han superado a las técnicas anteriores para múltiples tareas y objetivos de los sistemas de Procesamiento de Lenguaje Natural.

En el ámbito de Pregunta-Respuesta se emplea técnicas de análisis sin-

táctico, reconocimiento de entidades y más para caracterizar la pregunta (típicamente factuales o de hechos concretos) y sus subtipos (por sus términos interrogativos, sí o no) para obtener el foco de la pregunta y razonamientos implícitos entre la pregunta y la respuesta. Con esta metodología, es posible analizar las colecciones de Pregunta-Respuesta para compararlas por las dificultades que presentan y caracterizar los errores que comenten los Modelos Neuronales de Pregunta-Respuesta. A continuación se describe dichas técnicas de Procesamiento de Lenguaje Natural para el análisis y caracterización de las colecciones propuestas en este trabajo.

2.5.1 Etiquetadores léxicos y análisis sintáctico

El análisis morfológico y sintáctico permite obtener las clases gramaticales de las palabras (sustantivos, adjetivos, adverbios, verbos, pronombres, preposiciones, etc.) y sus etiquetas morfosintácticas, en inglés, *Part of Speech* (POS). Se dividen en dos subcategorías: cerradas y abiertas. Las clases cerradas son aquellas fijas e invariantes en el idioma, generalmente palabras funcionales y cortas que permiten tener estructuras en gramáticas (preposiciones, conjunciones, etc.). Por el contrario, las clases abiertas (sustantivos, verbos, adjetivos y adverbios), se amplían o disminuyen con el uso de la lengua.

Estos etiquetadores se enfrentan al problema de la ambigüedad gramatical: una misma palabra en distintas frases, contexto y posición en la oración tiene clase gramatical y etiqueta léxica distinta. Además, incluyen técnicas

de análisis morfológico para aportar el lema de una palabra. Para el entrenamiento de etiquetadores léxicos, se emplean corpus y etiquetas léxicas definidas en él, variando para cada idioma. En el caso de inglés, se sigue el formato del corpus de *Brown* y su evolución *Penn Treebank* (P. Marcus et al.). *Penn Treebank* está compuesto por más de 4,5 millones de palabras de inglés americano. Cuenta con 36 etiquetas, 12 de puntuación y símbolos y 18 sintácticas (ver ejemplos de etiquetas morfosintácticas en la tabla 2.14).

Los métodos del área han evolucionado desde los métodos basados en reglas, estocásticos (HMM) y máxima entropía (como *CoreNLP* del trabajo Qi et al. (2018)), a modelos de Aprendizaje Automático como árboles de decisión (*TreeTagger* (Schmid, 2013)), y Deep Learning como los disponibles en *SpaCy* (Honnibal and Montani, 2016-2021) y *Stanza* (Qi et al., 2020).

Etiquetas morfosintácticas	Ejemplos
WP VBZ DT	*What is the*, *Who is a*, *Who is another*
WDT IN DT	*Which of the*, *Which of these*, *Which in these*
WP VBD DT	*What was the*, *Who were the*, *What were those*
WP NN IN	*What kind of*, *What type of*, *What information about*
WP VVD DT	*What made the*, *What inspired the*, *Who played the*
IN WP NN	*In what country*, *For what movie*, *From what basis*
WP VVD NP	*What did Beyoncé*, *Who assisted Nepal*, *Whom did Sand*
VVG TO DT	*Mixing any to*, *According to some*, *Climbing to the*
WRB JJ NNS	*How many examples*, *When English people*, *When multiple pumps*
IN DT NN	*With the help*, *At the time*, *It the growth*

Tabla 2.14: *Etiquetas morfosintácticas de Penn Treebank (P. Marcus et al.) sobre ejemplos de palabras de comienzo de preguntas*

2.5.2 Reconocimiento de entidades

El reconocimiento de entidades o *Named Entity Recognition (NER)* consiste en la extracción de expresiones multipalabra que representan a personas, organizaciones, lugares (ciudades, localizaciones, estados, países, direcciones), cantidades (porcentajes, monetarias, cardianles y ordinales), referencias temporales (fechas, años, duración, tiempo, meses) y más. Los modelos avanzados también reconocen títulos de personas, ideologías, religiones, causas de crimen o de muerte y otras entidades (ver tabla 2.17 y desarrollo A.2). Para ello, se requiere un procesamiento previo de análisis sintáctico para la identificación de sustantivos propios, expresiones numéricas y cantidades, elementos de puntuación asociadas a unidades monetarias, porcentajes y más (Jurafsky and Martin, 2009-2021).

Los sistemas más básicos reconocen entidades en dominios específicos gracias al enriquecimiento con lexicones, diccionarios, tesauros y recursos disponibles online, para asociar a las entidades palabras y expresiones multipalabra de ámbitos generales; formación de reglas, con el etiquetado léxico y análisis sintáctico de corpora anotados y recursos previos del área. Otras librerías y servicios contrastan entidades del texto con las obtenidas de Wikipedia y más recursos digitales semánticos, como ontologías y redes semánticas como *WordNet* (Sarawagi, 2008). Actualmente, se plantea el reconocimiento de entidades como anotación de secuencias de texto con modelos estadísticos, *CoreNLP* (Manning et al., 2014), (Qi et al., 2018), y recientemente el uso de Modelos Neuronales de Lenguaje pre-entrenados incorporados en las

librerías.

Dentro de este planteamiento se cuenta con las librerías *SpaCy* (Honnibal and Montani, 2016-2021), *Flair* y *Stanza*, versión actualizada de *CoreNLP* con representaciones basadas en modelos neuronales para cada fase para mejor representación de las palabras en su contexto, ayudando también a su desambiguación y sentido (Qi et al., 2020) (ver comparativa de la tabla 2.15).

Language	Corpus	# Types	Stanza	FLAIR	spaCy
Arabic	AQMAR	4	**74.3**	74.0	–
Chinese	OntoNotes	18	**79.2**	–	–
Dutch	CoNLL02	4	89.2	**90.3**	73.8
	WikiNER	4	**94.8**	**94.8**	90.9
English	CoNLL03	4	92.1	**92.7**	81.0
	OntoNotes	18	88.8	**89.0**	85.4*
French	WikiNER	4	**92.9**	92.5	88.8*
German	CoNLL03	4	81.9	**82.5**	63.9
	GermEval14	4	85.2	**85.4**	68.4
Russian	WikiNER	4	**92.9**	–	–
Spanish	CoNLL02	4	**88.1**	87.3	77.5
	AnCora	4	**88.6**	88.4	76.1

Tabla 2.15: *Eficiencia obtenida para reconocimiento de entidades con Stanza, Flair y SpaCy (Qi et al., 2020)*

Como se observa en la figura 2.16, otra de las ventajas de los Modelos Neuronales de Lenguaje es que devuelven expresiones multipalabra basadas en sustantivos propios, comunes, en conjunto con preposiciones y conjunciones (*the Tower District*), números y elementos de puntuación (*8,868 tweets per second, one-half mile*).

Tabla 2.16: *Ejemplos de anotación NER con SpaCy sobre documentos de SQuAD*

Entidad	*CoreNLP*	*SpaCy*	*Stanza*
PERSON	Sí	Sí	Sí
ORGANIZATION	Sí	Sí (ORG)	Sí (ORG)
CITY	Sí	Sí (GPE)	Sí (GPE)
COUNTRY	Sí	Sí (GPE)	Sí (GPE)
LOCATION	Sí	Sí (LOC)	Sí (LOC)
NATIONALITY	Sí	Sí (NORP)	Sí
DATE	Sí	Sí	Sí
DURATION	Sí	Sí	Sí (TIME)
TIME	Sí	Sí	Sí
PERCENT	Sí	Sí	Sí
NUMBER	Sí	Sí (CARDINAL)	Sí (CARDINAL)
ORDINAL	Sí	Sí	Sí
MONEY	Sí	Sí	Sí
LANGUAGE	Sí	Sí	Sí
IDEOLOGY	Sí	Sí (NORP)	Sí (NORP)
RELIGION	Sí	Sí (NORP)	Sí (NORP)
STATE OR PROVINCE	Sí	Sí (GPE)	Sí
CRIMINAL CHARGE	Sí	No	No
CAUSE OF DEATH	Sí	No	No
MISC	Sí	No	Sí
URL	Sí	No	No
HANDLE	Sí	No	No
TITLE	Sí	No	No
SET	Sí	No	No
PRODUCT	No	Sí	Sí
FAC	No	Sí	Sí
LAW	No	Sí	Sí
WORK OF ART	No	Sí	Sí
EVENT	No	Sí	Sí
QUANTITY	No	Sí	Sí

Tabla 2.17: *Tipos de entidades disponibles en CoreNLP, spaCy y Stanza*

2.5.3 Obtención del foco de la pregunta

En los sistemas de Pregunta-Respuesta, las preguntas a contestar son generalmente sobre hechos concretos, resumen u opinión (Vicedo, 2004). En las colecciones de Búsqueda de Respuestas valoradas con sistemas basados en Modelos Neuronales de Lenguaje generalmente se trabaja con las preguntas de hechos concretos (*factoid questions* en inglés) que tienen respuesta en el contexto o fragmento seleccionado (pregunta-respuesta extractivo). El foco de la pregunta, en forma de palabra o conjunto de palabras, permite contextualizar y desambiguar el lenguaje para facilitar la obtención de la respuesta con este planteamiento (Martínez-Barco et al., 2014).

El análisis de las preguntas con técnicas de análisis sintáctico permite definir el foco de la pregunta y así asociar el tipo de respuesta o entidad que se requiere de forma implícita o explícita, como ejemplo, ver 2.19.

En el caso implícito, en función de taxonomías o clases semánticas predefinidas con el conocimiento general (Martínez-Barco et al., 2014) se identifica el tipo de respuesta esperada. En dicho análisis también se incluye la identificación de la partícula interrogativa de la pregunta (en inglés términos *wh*: *where*, *who*, *when*, etc.) y el reconocimiento de entidades de la respuesta. Como se puede ver en la tabla de ejemplos 2.20, *when* indica que la pregunta tiene como respuesta una expresión temporal, fecha o tiempo, incluidas en la taxonomía de la figura 2.18.

Q-class	Q-subclass	A-type
WHAT	basic what what-who what-when what-where	money | number | definition | title | nnp | undefined person | organization date location
WHO		person | organization
HOW	basic how how-many how-much how-far how-tall how-rich how-large	Maner number money | price distance number undefined number
WHERE		Location
WHEN		Date
WHICH	which-who which-where which-when which-what	Person location date nnp | organization
NAME	name-who name-where name-what	person | organization location title | nnp
WHY		Reason
WHOM		person | organization

Tabla 2.18: *Resumen de la taxonomía de preguntas Moldovan (Moldovan et al., 2000), (Martínez-Barco et al., 2014)*

Pregunta	Foco
When was the Tower Theatre built?	*time*
Where was the Tower Theatre built?	*place*
Which name is also used to describe the Amazon rainforest in English?	*name*
Jay Z and Beyonce attended which event together in August of 2011?	*event*
In what country is Normandy located?	*country*

Tabla 2.19: *Ejemplos de pregunta, focos y tipo de entidad de la respuesta sobre ejemplos de SQuAD*

Sin embargo, en otros casos, el foco de la pregunta aparece explícito en la pregunta: se trata del primer sustantivo o el sustantivo núcleo de sintagmas nominales de la pregunta, por el principio de composición del lenguaje. Esto sucede frecuentemente en preguntas con la partícula *what*. Algunos de estos sustantivos obtenidos es un tipo de entidad a reconocer en el contexto.

Así, en los sistemas con análisis superficial de Procesamiento de Lenguaje Natural, se extrae el tipo de respuesta en función del foco y clases semánticas de los hechos, que requiere el reconocimiento de entidades, análisis sintáctico y el procesamiento de palabras de la pregunta con valores discriminatorios o palabras claves que permiten la localización de aquellos fragmentos susceptibles de contener la respuesta.

Foco	Partículas	Entidades de respuesta
time	*When*	DATE, TIME, DURATION
place	*Where*	CITY, COUNTRY, LOCATION, STATE OR PROVINCE
person	*Who*	PERSON, TITLE, NATIONALITY

Tabla 2.20: *Ejemplos de asociación entre foco de la pregunta y tipo de entidad de la respuesta (NER) para preguntas de hechos concretos*

Esta caracterización sintáctica y semántica del lenguaje también es aplicable para la validación de respuestas extraídas por sistemas más complejos, como los Modelos Neuronales de Lenguaje.

Capítulo 3

Caracterización de colecciones de Pregunta-Respuesta

Durante el estudio del estado del arte se ha podido observar que los estudios de Búsqueda de Respuestas proponen colecciones para evaluar distintos Modelos Neuronales con métricas de Aprendizaje Automático y una inspección y categorización reducida y heterogénea de los errores cometidos.

Ante la falta de un criterio unificado, este trabajo aporta un estudio de fenómenos lingüísticos que pueden darse de forma simultánea en los ejemplos de una colección como punto de partida de nuevos planteamientos de anotación y detección para sistemas de predicción de dificultad. Posteriormente, para observar estas dificultades de forma generalizada en este capítulo se propone la caracterización de la formulación del lenguaje de las colecciones

de forma homogénea mediante el análisis de sus preguntas y respuestas con técnicas de Procesamiento de Lenguaje Natural y análisis de datos.

3.1 Estudio de fenómenos lingüísticos

A continuación se enumera las principales observaciones, necesidades y retos disponibles en las colecciones de Pregunta-Respuesta contempladas en la sección 2.2. Todos ellos suponen dificultades para los Modelos Neuronales de Búsqueda de Respuestas y son fenómenos a tener en cuenta para el estudio de dificultad y siguientes mejoras del área.

1. La contextualización es errónea debido a escasez de términos comunes entre la pregunta, contexto y la respuesta (*partial clue*, *word/lexical/exact match*) por variaciones léxicas (sinonimia, antonimia, hiperonimia, hiponimia, etc.), analogías o paráfrasis.

2. En caso de existencia de términos comunes hay confusión al tratar con negaciones, correferencias (erróneas o no), cambio de entidades o existencia de múltiples entidades del mismo tipo.

3. Dificultad en la selección del comienzo o final de la respuesta (en el caso de Pregunta-Respuesta extractivo).

4. Se tiene estructura sintáctica complicada, información adicional o errónea que causa confusión. Es necesario robustez de los modelos ante modificaciones ruidosas (Aspillaga et al., 2020).

5. Fallo al obtener la respuesta a partir de múltiples oraciones, falta de capacidad de razonamiento lógico (en ciencia o matemáticas), análisis global del texto y síntesis (Clark and Etzioni, 2016).

6. Se requiere conocimiento externo, común o general, inferencia o entendimiento de opinión para responder (Vicedo, 2004). También, en base al pre-entrenamiento y adaptación en una colección concreta tener capacidad de obtención de altas métricas y generalización de conocimiento ante otra colección de Pregunta-Respuesta (Talmor and Berant, 2019).

7. Se realiza un tratado erróneo o distinto del lenguaje, bien en las tareas técnicas de análisis sintáctico, detección de entidades nombradas; o bien por errores ortográficos, extranjerismos o múltiples idiomas, variaciones morfológicas, manejo de siglas y acrónimos.

8. Hay dificultad en la selección del tipo de respuesta por su forma (sintagma, booleana, numérica) o tipo de entidad.

9. Es necesaria la elección de respuesta correcta entre varias aceptables presentes en el texto; en el caso de Pregunta-Respuesta multirespuesta distinguir la correcta de los distractores.

10. Presencia de ambigüedades o por ser preguntas muy complejas, que hacen dudar también a una persona (*ambiguous*, *insufficient*, *hard*).

11. No se tiene soporte ante sinonimia, analogías, paráfrasis, pragmática ni similitud semántica textual (Rogers et al., 2020).

12. Duda de si responder o no ante preguntas no contestables por su formulación con negaciones, antonimia, cambios de entidades, contradictorias (Rajpurkar et al., 2018) o solicitud de información no presente en el texto (esperable baja confianza del modelo).

En el anexo B.2 se muestran ejemplos de los fenómenos descritos. En ellos se demuestra que el juicio de dificultad para su categorización no es único ya que los fenómenos lingüísticos contemplados coexisten en sus textos, además de suponer un proceso costoso en esfuerzo manual y tiempo. Por ello, la caracterización lingüística y el análisis de las respuestas dadas por los modelos permite su detección, anotación y clasificación por dificultad de forma homogénea y generalizada.

3.2 Análisis por caracterización lingüística

Para el estudio de dificultad *a priori*, en esta sección se analiza la formulación del lenguaje de las preguntas, respuestas y textos de las colecciones de Pregunta-Respuesta tras el uso de técnicas de Procesamiento de Lenguaje Natural para la obtención de etiquetas morfosintácticas, entidades reconocidas y foco de la pregunta.

El análisis estadístico por etiquetas lingüísticas permite estudiar patrones a partir de la frecuencia de aparición. También, establecer métodos de caracterización y asociación automática entre el foco de la pregunta y entidad de

la respuesta para obtener nueva información y subgrupos de preguntas, respuestas y contextos para su comparativa en diversas colecciones. En concreto, se ha realizado esta caracterización sobre las colecciones *SQuAD*, *NewsQA* y *RACE* porque permiten observar variedad del lenguaje en función de la tarea de Búsqueda de Respuestas que proponen.

Como caracterización previa se analizan las longitudes de los textos para cada colección en la tabla 3.1. Se observa que para las colecciones de Pregunta-Respuesta extractivo (*SQuAD* y *NewsQA*) la longitud de la respuesta es menor con respecto a la colección de Pregunta-Respuesta multi-respuesta (*RACE*[1])

Colección	Pregunta *train*	Pregunta *test+dev*	Respuesta *train*	Respuesta *test+dev*	Contexto *train*	Contexto *test+dev*
SQuAD 2.0	**10,93**	**11,05**	3,24*	3,28*	124,76	133,2
NewsQA	7,6	7,59	4,22*	4,26*	**556,70**	**557,76**
RACE	10,56	10,08	**6,80**	**6,33**	288,91	256,68
RACE-M	9,65	9,54	5,49	5,57	206,02	208,21
RACE-H	10,92	10,40	7,33	6,77	322,66	285,16

Tabla 3.1: *Análisis de longitud media de las preguntas, respuestas (*no nulas) y contextos de las colecciones de Pregunta-Respuesta SQuAD (train, dev), NewsQA (contextos de CNN Daily Mail) y RACE para train, test y dev*

Respecto a los textos empleados en Búsqueda de Respuestas, la colección *CNN Daily Mail* (asociada a *NewsQA*) usa noticias que son más largas, en

[1]Para obtener la longitud de las respuestas de *RACE*, se ha considerado la longitud media teniendo en cuenta las palabras de todas las opciones y dividido por 4 por haber dicho número de opciones: 3 distractores y una respuesta correcta.

media, que los fragmentos de *Wikipedia* que se utilizan en *SQuAD*. Así, *NewsQA* y *CNN Daily Mail* retan más a los modelos dado que presentan entradas de texto más largas, ya que en el diseño de Modelos Neuronales de Lenguaje se ajusta la longitud de la entrada que reciben, truncando los textos si superan una determinada longitud que suele ser de 512 palabras (*tokens*).

A continuación se comparan en detalle preguntas y respuestas a partir del resultado del análisis sintáctico, reconocimiento de entidades y agrupaciones por la relación entre el foco de la pregunta y entidad esperada como respuesta.

3.2.1 Análisis de las preguntas

Al analizar el tipo de preguntas en inglés, se realiza la inspección por su formulación y presencia de los términos *wh* al comienzo: *where*, *who*, *when*, etc. que permite una primera clasificación de las preguntas por el tipo de respuesta que requiere.

En la tabla 3.2 se realiza una comparativa en cuanto a las partículas interrogativas. En el caso de *SQuAD* sus diez partículas más frecuentes al comienzo de la pregunta (en un 86,42 % del total de preguntas) son términos *wh* (*what*, *who*, *when*, *which*, *where*, *why*) y otras palabras *stopwords* (*the*, *in*, *how*), mientras que en *NewsQA* también se encuentran dentro de las diez palabras más frecuentes al comienzo los verbos *is* (991 preguntas en total,

0,9 %), *did* (0,75 %, 814 preguntas) y para *RACE* *according* (5,78 %, 5.533 preguntas), *from* (2,67 %, 2.558 preguntas), *we* (2,67 %, 2.562 preguntas).

Colección	*SQuAD 2.0* *train*	*SQuAD 2.0* *dev*	*NewsQA* *train*	*NewsQA* *test+dev*	*RACE* *all*
what	**44,63 % (58.174)**	**47,67 % (5.660)**	**44,06 % (47.448)**	**43,49 % (5.201)**	**16,83 % (5.201)**
who	9,43 % (12.299)	7,78 % (926)	**19,15 % (20.620)**	**19,92 % (2.382)**	1,34 % (1.278)
how	**8,76 % (11.421)**	**9,34 % (1.110)**	6,82 % (7.352)	6,92 % (835)	3,95 % (3.775)
when	5,92 % (7.724)	5,69 % (676)	4,14 % (4.465)	3,79 % (447)	2,56 % (2.448)
in	**4,95 % (6.453)**	**4,29 % (510)**	0,63 % (679)	0,69 % (83)	2,6 % (2.486)
which	4,31 % (5.624)	2,72 % (324)	2,16 % (2.329)	2,18 % (261)	**12,24 % (11.701)**
where	3,54 % (4.617)	3,42 % (407)	**7,07 % (7.616)**	**7,39 % (884)**	1,27 % (1.211)
the	2,37 % (3.090)	2,42 % (288)	0,82 % (878)	1,00 % (119)	**15,33 % (14.664)**
why	1,41 % (1.833)	1,53 % (182)	0,10 % (102)	0,10 % (12)	**4,68 % (4.477)**
Otras	14,05 % (18.321)	14,71 % (985)	15,03 % (16.185)	14,50 % (1.735)	**39,21 % (37.496)**

Tabla 3.2: *Comparativa por palabra de comienzo de las preguntas de las colecciones de Pregunta-Respuesta SQuAD, NewsQA y RACE*

La tarea de opción múltiple en *RACE* afecta a cómo se formulan las preguntas ya que se reduce el uso de partículas *wh* al contextualizar y referenciar la información del texto para la selección o continuación de la oración en las respuestas.

3.2.1.1 Análisis sintáctico de las preguntas

En esta sección se presenta un análisis más general de la formulación de las preguntas al utilizar análisis sintáctico y para ello se obtiene grupos de preguntas por su formulación. Concretamente, por su estructura sintáctica completa se obtiene unas particiones con pocos ejemplos por cada una (ver la tabla 3.3) lo que denota variabilidad en la formulación de las preguntas.

Para formar grupos más significativos se emplea a continuación una ca-

Colección	*SQuAD 2.0 train*	*SQuAD 2.0 dev*	*NewsQA train*	*NewsQA test+dev*	*RACE all*
WP VVZ NP VV IN	212	4	66	11	9
WP VBD NP POS NN	94	14	58	14	6
WP VVD DT NN	55	8	538	64	24
WP VBZ NP NP	40	2	492	58	36
WP VBZ DT JJS - NN IN DT NN	27	1	6	1	1.181
Otras	129.891 (99,67 %)	11.844 (99,75 %)	106.514 (98,92 %)	11.811 (98,76 %)	94.369 (98,68 %)

Tabla 3.3: *Comparativa por estructura sintáctica de toda la pregunta de las colecciones de Pregunta-Respuesta SQuAD, NewsQA y RACE*

racterización lingüística por trigramas morfosintácticos del comienzo de la pregunta. Como se observa en la tabla 3.4, los trigramas de etiquetas morfosintácticas se distribuyen de forma similar en las colecciones de Pregunta-Respuesta extractivo (*SQuAD* y *NewsQA*). En todas las colecciones destaca la estructura *WP VBZ DT* (por ejemplo *What is the...*). Además, este trigrama se divide en los cuatrigramas más frecuentes seguido de un sustantivo común, generalmente el foco de la pregunta (*WP VBZ DT NN*), o un adjetivo (*WP VBZ DT JJ*).

Con el análisis de sus frecuencias, se observa que ciertas secuencias de trigramas morfosintácticos son más característicos que otros en ciertas colecciones: *WDT IN DT* en un $9,38\,\%$ de las preguntas de *RACE*, (por ejemplo *Which of the...*), *IN WP NN* en un $4,36\,\%$ de casos de *SQuAD train* (por ejemplo *In what country...*) y *WP VVZ NP* y *WP VBZ NP* en un $2,22\,\%$ y $2,46\,\%$, respectivamente, de *NewsQA train* (*What does Rosen...*), destacadas en la tabla 3.4 con ejemplos asociados en la tabla 2.14.

Etiquetas	*SQuAD 2.0 train*	*SQuAD 2.0 dev*	*NewsQA train*	*NewsQA test+dev*	*RACE all*
WP VBZ DT	8.314	925	5.924	653	5.446
WDT IN DT	45	2	5	1	**8.971**
WP VBD DT	5.986	383	2.994	325	331
WP NN IN	5.579	515	1.858	192	334
WP VVD DT	4.858	401	5.806	627	982
IN WP NN	**5.679**	440	796	85	157
WP VVD NP	3.592	286	**4.208**	497	602
VVG TO DT	130	10	53	2	**4.792**
WRB JJ NNS	3.883	356	3.759	426	1.017
IN DT NN	522	38	62	9	**4.070**
PP MD VV	2	0	16	6	2.401
WP MD PP	45	4	303	36	2.056
WP NN VVD	4.080	239	1.307	123	103
WP VBD VVN	1.306	84	2.481	290	21
WP VVZ DT	1.891	258	2.390	305	1.581
WP VBZ NP	814	61	**2.233**	262	536
WRB VVD NP	2.784	248	844	82	1.108
WP VVZ NP	881	62	**2.012**	241	580
WRB VBD DT	2.747	254	1.813	226	298
DT NN VVZ	58	3	87	15	1.946
DT JJ NN	546	55	139	10	1.934
DT NN VBZ	60	2	109	14	1.883
DT NN IN	737	71	129	11	1.744
Otros	75.780 (58,14 %)	7.176 (60,43 %)	68.346 (63,47 %)	7.521 (62,88 %)	52.732 (55,14 %)

Tabla 3.4: *Comparativa de número de preguntas por trigrama de etiquetas morfosintácticas al comienzo de las preguntas de las colecciones de Pregunta-Respuesta SQuAD, NewsQA y RACE*

Estas agrupaciones denotan diferencias en la formulación de las preguntas de las colecciones y motiva la caracterización del foco de la pregunta presente al comienzo en forma de sustantivos comunes, *NN* o *NNS*, precedidos por determinantes (*DT*) versus a la formulación con a referencias entidades a partir de nombres propios (uso más frecuente en NewsQA).

3.2.1.2 Análisis del foco de las preguntas

A partir del análisis sintáctico se extrae que los focos más frecuentes en las colecciones de Pregunta-Respuesta extractivo son *person* (asociado a la partícula *who*), *time* (asociado a la partícula *when*), *place* (asociado a la partícula *where*) y otros en base al análisis de sustantivos: *year*, *name*, *country*, *kind*, *number* y *percentage* (más detalle en la tabla 3.5).

Sin embargo, para *RACE*, los focos más comunes son *passage* (*What can we infer from the passage?*), *following* (*What of the following is NOT true?*), *author* (*The author writes this note to...*), *writer* (*The writer of The Little Prince is from...*), *title* (*What is the best title for the text?*). Por el tipo de lenguaje, las preguntas y focos *statements* (*What of the following statements is TRUE?*) y *purpose* (*The purpose of the text is to introduce...*) denotan la tarea de multirespuesta, preguntas con inferencia y de opinión sobre el texto.

Colección	*SQuAD 2.0* *train*	*SQuAD 2.0* *dev*	*NewsQA* *train*	*NewsQA* *test+dev*	*RACE* *all*
person	9,81 % (12.779)	8,46 % (1.005)	19,42 % (20.906)	20,12 % (2.406)	1,56 % (1.491)
number	7,87 % (10.261)	0,31 % (37)	0,81 % (872)	0,71 % (85)	0,01 % (95)
time	6,31 % (8.229)	6,53 % (775)	4,34 % (4.673)	3,95 % (472)	2,80 % (2.676)
place	3,70 % (4.823)	3,66 % (434)	7,17 % (7.725)	7,46 % (892)	1,37 % (1.306)
year	3,34 % (4.351)	2,46 % (292)	0,40 % (432)	0,34 % (41)	0,08 % (81)
name	2,09 % (2.727)	1,89 % (224)	1,13 % (1.217)	1,21 % (145)	0,05 % (50)
type	1,82 % (2.377)	2,23 % (265)	0,38 % (406)	0,34 % (41)	0,04 % (43)
people	0,31 % (401)	0,70 % (83)	1,38 % (1.481)	1,44 % (172)	1,61 % (1.540)
country	0,70 % (917)	0,67 % (80)	0,71 % (768)	0,82 % (98)	0,10 % (97)
percentage	0,74 % (968)	0,66 % (78)	0,18 % (196)	0,25 % (30)	0,02 % (15)
kind	0,74 % (967)	0,83 % (99)	0,53 % (574)	0,48 % (58)	0,35 % (337)
age	0,14 % (177)	0,08 % (10)	0,56 % (600)	0,51 % (61)	0,02 % (22)
Otras	62,41 % (81.344)	71,51 % (8.491)	62,99 % (67.827)	62,36 % (7.458)	91,89 % (87.872)

Tabla 3.5: *Comparativa por foco de la pregunta de las colecciones de Pregunta-Respuesta SQuAD, NewsQA y RACE*

3.2.1.3 Análisis de coincidencias entre preguntas y contextos

Los fenómenos lingüísticos más evaluados en el estudio del estado del arte son relativos a la variabilidad lingüística que tiene la pregunta con respecto al texto para contextualización del fragmento de respuesta. Para ello, las técnicas de n-gramas, coincidencias de palabras y estructuras (*chunking*) y más avanzadas de similitud textual permiten comparar entre estructuras sintácticas y proximidad de las palabras para realizar un análisis pormenorizado. Estos efectos denominados *word matching*, *lexical match* o *partial clues* en inglés, son similares a los que realiza una persona al contextualizar la información y al analizar un texto (diferentes lecturas preliminar o detallada como subrayar) de un texto para su análisis.

Una mayor coincidencia de los términos de la pregunta en los textos ge-

neralmente facilita la contextualización y búsqueda de la respuesta, mientras que el uso de otras palabras como por ejemplo sinónimos, añade dificultad. Con un procesado automático superficial, en este apartado se ha analizado en forma de número de coincidencias (palabras y entidades reconocidas) comunes en la formulación de la pregunta y texto del que extraer la respuesta. En media, se tiene unas $6,6$ palabras coincidentes entre pregunta y contexto en *SQuAD*, mientras que en *NewsQA* se tiene $5,79$ en común en media.

Con el análisis de las entidades anotadas en las preguntas, en esta sección se observa que la colección de *SQuAD* tiene mayor número de entidades que permiten contextualizar el foco y la información a extraer del texto como respuesta como se observa al comparar las entidades más frecuentes de los fragmentos de la colección con respecto a la entidades frecuentes de las preguntas (ver resumen en la figura 3.1 y más detalle en el anexo, tabla B.1).

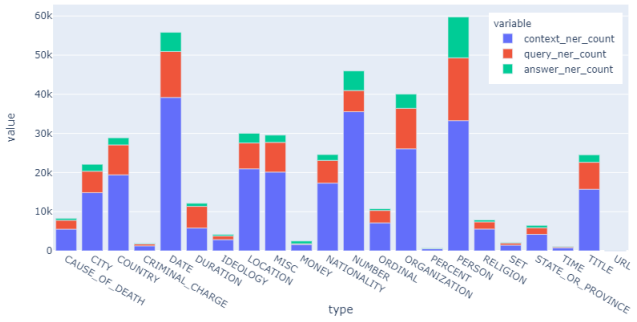


Figura 3.1: *Número de entidades anotadas en SQuAD v2.0 train en las preguntas, contextos y respuestas*

Entidad	Preguntas	Contextos
SQuAD *PERSON*	*John*, *Napoleon*, *Nasser*, *Eisenhower*, *Jesus*, *Christ*, *Maddona*, *Mary*, *Chopin*	*Chopin*, *Napoleon*, *Eisenhower*, *Beyonce*, *Elizabeth*, *Kanye* *Nasser*, *Beyoncé*, *Madonna*
NewsQA *PERSON*	***Obama***, *Bush*, ***Clinton***, *Jackson*, ***Barak Obama***, *McCain* *Brown*, *Michael Jackson*, *Hilary Clinton*, *Smith*	*Obama*, *Bush*, *Clinton*, *Jackson*, *McCain*, *Barak Obama* *Michael Jackson*, *Kennedy*, *Hillary Clinton*, *Palin*
RACE *ORGANIZATION*	*Facebook*, *Apple*, *Google*, *NASA*, *BBC*, *Harvard*, *McDonald*, *Disney*, *NBA*, *Microsoft*	*Google*, *Apple*, *NASA*, *McDonald*, *Para*, *Facebook* *Disney*, *Browns*, *Alibaba*, *Harvard*
SQuAD *COUNTRY*	***United States***, ***U.S.***, *France*, *China*, *India*, *England* *Germany*, *America*, *Britain*, ***US***	*US*, *France*, *United States*, *England*, *China*, *America* *Britain*, *U.S.*, *UK*, *India*

Tabla 3.6: *Top 10 de entidades reconocidas de las preguntas y en los textos de las colecciones de Pregunta-Respuesta en entrenamiento*

A partir de los ejemplos de la tabla 3.6 se puede observar la variabilidad léxica en entidades detectadas (*United States*, *US* y *U.S.*) y fallos ortográficos (*Beyonce*, *Beyoncé*), más relaciones semánticas por analogía (*America*, *US* y *Britain*, *UK*), omisiones y correferencias (*Obama*, *Barak Obama*).

3.2.2 Análisis de las respuestas

Las respuestas a las preguntas factuales en Pregunta-Respuesta extractivo están caracterizadas por el tipo de entidad y su estructura sintáctica, observando relación con las proporciones de entidades más solicitadas en las preguntas (focos de las preguntas más frecuentes). En cuanto a su variedad lingüística, en la siguiente subsección se analiza a gran escala las colecciones de Pregunta-Respuesta extractivo.

3.2.2.1 Análisis sintáctico de las respuestas

Realizando un análisis superficial a partir del análisis sintáctico de las respuestas se obtiene la siguiente proporción de respuestas por formulación de mayor a menor frecuencia (ver tabla 3.7): nombres propios (*NP, NPS*), sustantivas (*NN, NNS*), sintagmas adjetivales (*JJ*), numéricas (*CD*), verbales y adverbiales (*RB, RBR, RBS*).

Colección	*SQuAD 2.0 train*	*SQuAD 2.0 dev*	*NewsQA train*	*NewsQA test+dev*
Nombres propios	27,06 % (35.269)	17,35 % (2.060)	37,41 % (40.276)	38,2 % (4.568)
Sustantivas	11,14 % (14.517)	11,35 % (9,56)	9,18 % (9.880)	9,23 % (1.104)
Adjetivales	14,23 % (18.547)	12,32 % (1.463)	15,2 % (16.364)	15,06 % (1.801)
Numéricas	5,39 % (7.026)	3,71 % (440)	6,29 % (6.775)	6,14 % (734)
Verbales	2,68 % (3.499)	2,86 % (339)	6,06 % (6.527)	5,84 % (699)
Adverbiales	0,63 % (819)	0,69 % (82)	0,8 % (864)	0,76 % (261)
Otras*	38,86 % (50.642)	53,51 % (6.354)	25,06 % (26.988)	24,76 % (2.962)

Tabla 3.7: *Comparativa por formulación de las respuestas las colecciones de Pregunta-Respuesta SQuAD y NewsQA, *incluyendo respuestas a preguntas null*

3.2.2.2 Análisis de las entidades de las respuestas

Desde un punto de vista de anotación global, se compara la proporción de entidades total reconocidas en las respuestas para cada colección y para cada tipo. Como se ver en la tabla B.2, las entidades de respuesta más frecuentes en todas las colecciones son *PERSON* y *NUMBER* (focos de la pregunta *number* y *year*). En tercera posición se encuentra *DATE* (relacionada con el foco *time* y *year*) en el caso de *SQuAD*, *ORGANIZATION* en el caso de

NewsQA y *TITLE* para *RACE*. Se dispone de más detalle en el anexo para su comparativa.

A continuación se detalla cómo esas entidades se distribuyen en las respuestas a extraer, teniendo en cuenta que hay preguntas que contienen múltiples entidades en ellas, ver tabla 3.8. En otros casos, no se ha anotado entidades por ser respuestas vacías (preguntas *null*) o por no haber entidad reconocida en ella.

Colección	*SQuAD 2.0* *train*	*SQuAD 2.0* *dev*	*NewsQA* *train*	*NewsQA* *test+dev*
PERSON	**8,58 % (11.192)**	**5,62 % (668)**	**8,52 % (9.181)**	**8,78 % (1.050)**
ORGANIZATION	2,52 % (3.291)	1,80 % (214)	1,86 % (2.005)	1,70 % (204)
TITLE	0,95 % (1.240)	0,65 % (78)	0,73 % (792)	0,81 % (98)
NATIONALITY	1,25 % (1.639)	0,67 % (80)	0,77 % (829)	0,92 % (111)
RELIGION	0,61 % (800)	0,08 % (9)	0,09 % (106)	0,16 % (20)
IDEOLOGY	0,33 % (441)	0,40 % (48)	0,17 % (189)	0,16 % (20)
COUNTRY	1,49 % (1.946)	0,98 % (117)	2,11 % (2.278)	2,33 % (279)
LOCATION	1,04 % (1.364)	0,86 % (103)	0,57 % (619)	0,61 % (73)
CITY	1,17 % (1.531)	0,76 % (91)	0,83 % (895)	0,89 % (107)
DATE	**7,05 % (9.212)**	**4,13 % (491)**	**2,69 % (2.901)**	**0,11 % (293)**
TIME	0,05 % (78)	0,02 % (3)	0,63 % (68)	0,11 % (14)
DURATION	0,61 % (798)	0,00 % (0)	1,03 % (1.110)	0,79 % (95)
NUMBER	**7,10 % (9.255)**	**4,83 % (574)**	**5,85 % (6.299)**	**5,98 % (716)**
PERCENT	0,10 % (131)	0,03 % (4)	0,19 % (208)	0,29 % (35)
MONEY	0,54 % (712)	0,34 % (41)	0,87 % (936)	0,98 % (118)
ORDINAL	0,59 % (769)	0,29 % (35)	0,24 % (262)	0,19 % (23)
LANGUAGE	0,00 % (0)	0,00 % (0)	0,00 % (0)	0,00 % (0)
STATE OR PROVINCE	0,31 % (410)	0,12 % (15)	0,59 % (644)	0,61 % (73)
CRIMINAL CHARGE	0,12 % (168)	0,12 % (15)	0,46 % (497)	0,50 % (60)
CAUSE OF DEATH	0,41 % (532)	0,34 % (41)	0,84 % (910)	0,83 % (100)
SET	0,25 % (329)	0,18 % (71)	0,04 % (48)	0,06 % (8)
URL	0,01 % (20)	0,00 % (0)	0,05 % (53)	0,04 % (5)
MISC	1,01 % (1.323)	0,59 % (71)	0,31 % (333)	0,35 % (42)
HANDLE	0,00 % (0)	0,00 % (0)	0,002 % (3)	0,00 % (0)
Múltiple	**5,29 % (6.901)**	**3,48 % (414)**	**6,83 % (7.363)**	**7,08 % (847)**
Sin entidades	25,13 % (32.759)	22,94 % (2.724)	17,34 % (18.681)	17,53 % (2.071)
Sin respuesta	33,37 % (43.498)	50,07 % (5.945)	46,86 % (50.464)	45,96 % (5.497)

Tabla 3.8: *Comparativa por entidades reconocidas en las respuestas de Pregunta-Respuesta SQuAD y NewsQA*

Para ver variabilidad y cambios de entidades del contexto que el modelo debe manejar, se analiza el número medio de entidades del contexto coincidentes con el tipo esperado como respuesta, de forma que es posible medir dificultad en la selección de la respuesta correcta sobre dichas preguntas (ver tabla 3.9).

Colección	*SQuAD 2.0 train*	*SQuAD 2.0 dev*	*NewsQA train*	*NewsQA test+dev*
PERSON	**4,21**	**3,97**	**22,89**	**22,54**
ORGANIZATION	**4,15**	**5,49**	**15,51**	**14,57**
TITLE	3,06	2,98	11,21	9,94
NATIONALITY	3,83	2,87	8,36	8,70
RELIGION	3,59	3,02	7,41	8,30
IDEOLOGY	2,20	3,25	8,18	3,55
COUNTRY	4,19	3,58	14,59	15,39
LOCATION	3,90	4,09	8,77	7,10
CITY	3,21	3,48	7,59	7,91
DATE	3,89	4,00	12,02	11,64
TIME	2,23	1,33	4,51	4,14
DURATION	1,84	1,56	5,34	4,83
NUMBER	**5,18**	**6,16**	**11,71**	**11,84**
PERCENT	2,58	2,25	4,85	5,42
MONEY	2,62	3,39	6,03	6,27
ORDINAL	1,78	1,71	4,84	4,34
STATE OR PROVINCE	2,95	2,80	7,72	7,49
CRIMINAL CHARGE	2,48	2,26	5,70	6,18
CAUSE OF DEATH	2,95	3,65	10,09	9,11
SET	0,62	0,36	2,39	1,25
URL	1,35	0,00	4,64	1,40
MISC	3,46	2,66	7,06	5,80
HANDLE	0,00	0,00	9,66	0,00

Tabla 3.9: *Comparativa por número medio de entidades en los textos de Pregunta-Respuesta del mismo tipo a la entidad esperada como respuesta para SQuAD y NewsQA*

Por ejemplo, para la pregunta *When was the Tower Theatre built?* es posible contestar con distintas entidades del tipo *DATE* (*1939, today, 1916, now*) y es necesario más información para desambiguar y contextualizar la

respuesta, en este caso el verbo *built* aparece en el fragmento (ver figura 2.16 y el desarrollo en el anexo).

3.2.3 Análisis por asociación entre foco de la pregunta y tipo de respuesta

Es posible realizar una primera comprobación automática de primer nivel mediante la verificación de relaciones entre el foco de la pregunta y respuesta válida en base a reglas de asociación y taxonomías (Moldovan et al., 2000). Para ello, se emplean palabras literales del foco y su asociación con las etiquetas de entidades reconocidas, como ejemplifica la tabla 2.20 y la figura 2.18. Además se ha enriquecido con ciertos sinónimos de conocimiento general (ver el desarrollo A.5).

Se obtienen clases por el foco explícito encontrado en la pregunta que denota el nombre del tipo de entidad solicitada (*person* o *people* relaciona a entidad *PERSON*, clase *FNER-PERSON*), clases con información solicitada por el término o forma interrogativa (*when*, *where*, *how many*) que asocia un tipo de entidad en la respuesta de forma más o menos ambigua (casos de preguntas con partícula *what* generalmente), como se observa en la clase *FNER-PERSON-ORG*.

A partir de los resultados de la tabla 3.10, se detecta preguntas más complejas por semántica, coreferencias y variabilidad del lenguaje ante anologías u omisiones. En dichos casos la pregunta no es explícita al detallar la enti-

dad que debe contener la respuesta, pero la respuesta sí contiene una entidad nombrada detectada, señalada como *Otras*. También, en los casos en los que la respuesta no incluye entidad nombrada o no ha podido ser reconocida por su complejidad, se detalla como *No aplica*.

Colección	*SQuAD 2.0 train*	*SQuAD 2.0 dev*	*NewsQA train*	*NewsQA test+dev*
FNER-PERSON	3,97 % (5.175)	2,29 % (272)	**10,06 % (10.830)**	**10,64 % (1.274)**
FNER-ORGANIZATION	0,08 % (107)	0,08 % (10)	0,03 % (37)	0,04 % (5)
FNER-PERSON-ORG	**2,54 % (3.309)**	**2,20 % (261)**	**4,65 % (5.011)**	**4,56 % (545)**
FNER-TITLE	0,14 % (181)	0,00 % (1)	0,08 % (1)	0,00 % (0)
FNER-NATIONALITY	0,05 % (62)	0,04 % (5)	0,07 % (72)	0,07 % (8)
FNER-RELIGION	0,08 % (107)	0,08 % (9)	0,01 % (13)	0,03 % (3)
FNER-IDEOLOGY	0,01 % (11)	0,00 % (0)	0,00 % (1)	0,00 % (0)
FNER-LOC	1,79 % (2.328)	1,32 % (157)	4,48 % (4.827)	4,39 % (525)
FNER-COUNTRY	0,36 % (469)	0,69 % (82)	0,50 % (543)	0,59 % (70)
FNER-LOCATION	0,01 % (17)	0,02 % (2)	0,01 % (7)	0,00 % (0)
FNER-CITY	0,24 % (309)	0,19 % (22)	0,14 % (148)	0,15 % (18)
FNER-DATE	0,03 % (389)	0,04 % (5)	0,09 % (93)	0,08 % (9)
FNER-TIME	**6,14 % (8.004)**	**3,82 % (453)**	4,00 % (4.305)	3,56 % (426)
FNER-DURATION	0,00 % (2)	0,00 % (0)	0,00 % (4)	0,00 % (0)
FNER-NUMBER	5,58 % (7.272)	2,48 % (295)	5,22 % (5.624)	5,49 % (656)
FNER-PERCENT	0,01 % (13)	0,01 % (1)	0,01 % (9)	0,01 % (1)
FNER-MONEY	0,0 % (1)	0,00 % (0)	0,01 % (103)	0,08 % (10)
Otras	20,17 % (26.285)	14,13 % (1.678)	**25,20 % (27.136)**	**25,44 % (3.042)**
No aplica*	58,53 % (76.278)	73,01 % (8.669)	45,42 % (48.903)	44,88 % (5.367)

Tabla 3.10: *Comparativa por asociación del foco de la pregunta y tipo de respuesta por sus entidades reconocidas de colecciones de Pregunta-Respuesta SQuAD y NewsQA, *incluyendo respuestas a preguntas null o sin entidades reconocidas*

3.3 Análisis por resultados obtenidos con los modelos

A partir de la caracterización lingüística y los planteamientos actuales de Modelos Neuronales de Lenguaje en esta sección se propone los siguientes análisis de dificultad a gran escala para comparar fortalezas de cada modelo sobre una colección de Pregunta-Respuesta en particular. Para ello, se realiza la comparación de la respuesta dada para cada ejemplo de una colección de Pregunta-Respuesta al lanzar los modelos con la anotada en la colección, teniendo en cuenta leves diferencias entre ellas por puntuación y determinantes, ver el desarrollo A.7.

Para el análisis se han seleccionado modelos con especialización en la colección para la que estudiar su dificultad por caracterización lingüística en sus versiones públicas (entrenamiento, desarrollo y de evaluación o *test* si se dispone) para ver dificultades que suponen en global todas las preguntas. Además, para los modelos con el mismo entrenamiento en la colección concreta, el análisis de resultados se realiza comparando automáticamente si los Modelos Neuronales de Lenguaje responden de forma similar o idéntica a la respuesta considerada por los *crowdworkers*. Este método marca como errores aquellas respuestas que sean distintas a las respuestas de referencia de la colección, considerada como correcta[2]. Es decir, se recopilan todos los casos

[2]en ciertos casos se encuentran erratas, múltiples respuestas válidas o casos difíciles que hacen dudar a las personas implicadas en el proceso de construcción de la colección, ver ejemplo B.2.10 en el anexo.

en los que se difiere en la respuesta y se consideran como error del modelo para analizar si tienen caracterizaciones similares o si contienen un fenómeno lingüístico superado o no por una persona, de modo que es esperable que el desacuerdo implique mayor dificultad.

En concreto, se han seleccionado la colección de *SQuAD* y los modelos pre-entrenados de BERT (*base*, *large*), RoBERTa (*base*, *large*) para *question-answering* y T5 (*base*) para generación de la respuesta. Generalmente, para cada uno de estos modelos se tiene una versión adaptada sobre cada una de las colecciones[3]. En la tabla 3.11 se pueden ver los nombres y los resultados de estos modelos reportados oficialmente.

Modelo	Fine-tunning	Modelos	F1-score
BERT	*SQuAD 2.0*	phiyodr/bert-base-finetuned-squad-v2	73.90
		phiyodr/bert-large-finetuned-squad-v2	79.2
RoBERTa	*SQuAD 2.0*	deepset/roberta-base-squad2	83.00
		phiyodr/roberta-large-finetuned-squad2	87.89
T5	*SQuAD 1.0*	valhalla/t5-base-squad	89.86

Tabla 3.11: *Modelos de HuggingFace para Pregunta-Respuesta con fine-tunning con SQuAD*

En la tabla 3.12 se muestra la tasa de error de cada modelo sobre *SQuAD*. Hay que tener en cuenta que la ejecución sobre la colección de entrenamiento se realiza sobre el mismo conjunto con el que se ha adaptado el modelo. De este modo se pretende ver si aún así se obtienen errores. En efecto, se observa que el error medio para la partición de entrenamiento es menor, un

[3]No se dispone de modelos *XLNet* para Pregunta-Respuesta públicos para *SQuAD*.

$3,87\,\%$, mientras que en la colección de desarrollo se asciende a un error medio de $10,4\,\%$. Analizando el error a nivel de ejemplo, se tiene que todos los modelos han respondido erróneamente a un $0,39\,\%$ de las preguntas, correctamente a un $88,84\,\%$ en el caso de la partición de entrenamiento de *SQuAD*; para la partición de desarrollo erróneamente a un $2,60\,\%$ de las preguntas y correctamente un $76,55$ (ver más detalle de errores comunes por combinaciones de modelos en el anexo, tabla B.3).

Modelo	*SQuAD 2.0 train*	*SQuAD 2.0 dev*
phiyodr/bert-base-finetuned-squad-v2	3,57 % (3.104)	11,94 % (708)
phiyodr/bert-large-finetuned-squad-v2	1,46 % (1.270)	9,92 % (588)
deepset/roberta-base-squad2	4,86 % (4.220)	9,07 % (538)
phiyodr/roberta-large-finetuned-squad2	4,87 % (4.231)	9,53 % (565)
valhalla/t5-base-squad	4,58 % (3.982)	11,52 % (683)

Tabla 3.12: *Comparativa por errores cometidos sobre preguntas no null por los modelos con especialización en la colección de Pregunta-Respuesta SQuAD*

Para el análisis de error por caracterización lingüística que se va a mostrar en las siguientes secciones, se ha tomado aquellos ejemplos en los que al menos algún modelo haya cometido error, de forma que para *SQuAD train* se tienen 9.686 preguntas contestables (que asciende a 35.227 considerando también preguntas *null*) y para *SQuAD dev* 1.390 preguntas contestables (de un total de 4.930 erróneas). Es decir, se propone la hipótesis de que si algún modelo ha fallado la pregunta, en ella consta algún fenómeno lingüístico a superar.

3.3.1 Análisis de errores por longitudes

El análisis de errores de los modelos por la distribución de errores por longitudes permite observar si cuanto mayor longitud del texto supone más dificultad para su procesamiento u extracción. Para ello, la gráfica tanto para la partición de entrenamiento (ver figura 3.2) como la de desarrollo de *SQuAD* (ver figura 3.4) permite observar que los errores cometidos por los modelos tienen una distribución análoga por longitud de la pregunta y respuesta a la distribución de las preguntas contestadas correctamente.

Considerando grupos de al menos diez preguntas en *SQuAD train* se tiene porcentaje de error entre el 10 % y 16 % sobre preguntas de longitud 4 palabras a 26 palabras. Sin embargo, en *SQuAD dev* se tiene porcentaje de error entre el 15 % y 58 % sobre preguntas de longitud del mismo rango, detectando mayor porcentaje de error en los extremos, es decir, para 4 palabras (52 % de error) y de 25 palabras (58 %).

Sin embargo, la longitud de la respuesta a extraer sí conlleva más dificultad, ya que crece el número de palabras a extraer del texto, como se puede ver en las figuras 3.2 y 3.3. Se observa que para el conjunto de entrenamiento a partir de 11 palabras el error supera el 30 %, el mismo porcentaje se supera en el caso de *dev* a partir de las 6 palabras a extraer.

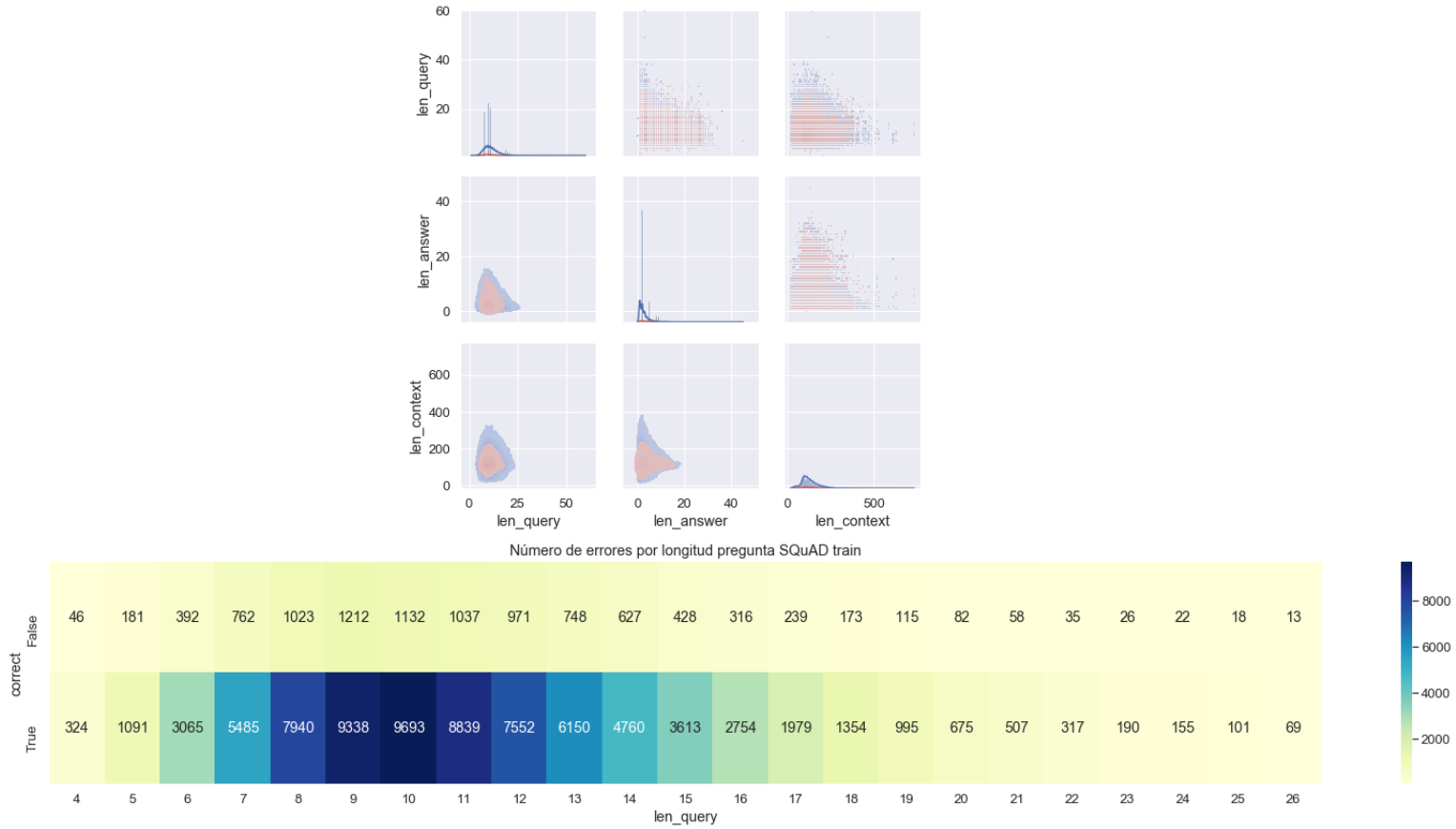


Figura 3.2: *Relación entre la longitud de la pregunta, respuesta y texto para SQuAD v2.0 train*

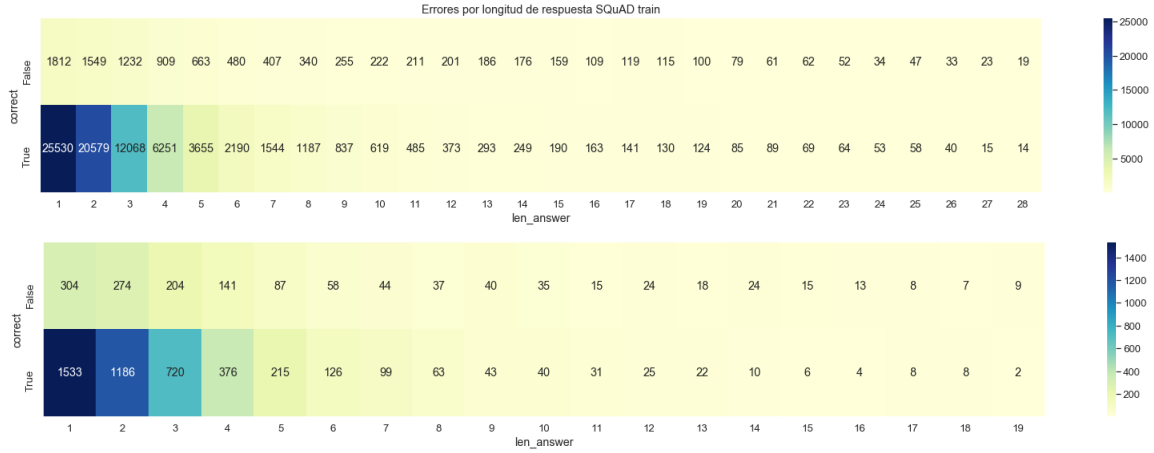


Figura 3.3: *Relación entre la longitud de la respuesta para SQuAD v2.0 train y dev*

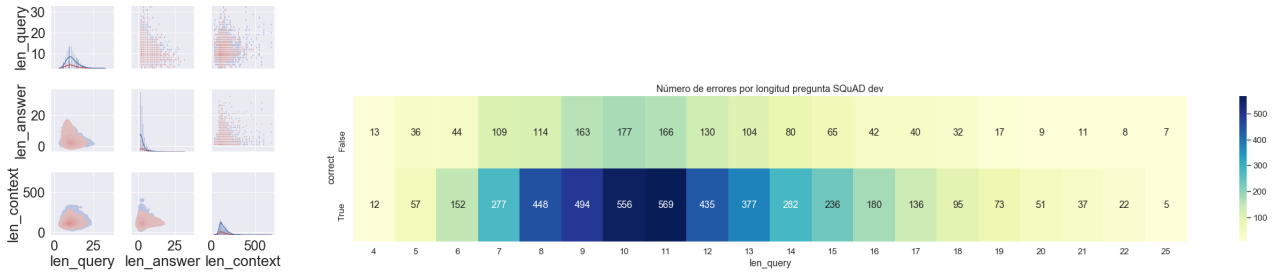


Figura 3.4: *Relación entre la longitud de la pregunta, respuesta y texto para SQuAD v2.0 dev*

3.3.2 Análisis de errores por tipos de preguntas

En este apartado se realiza el análisis de errores de los modelos por grupos de preguntas por su caracterización lingüística resultante del análisis sintáctico (partícula interrogativa y trigramas morfosintácticos) y técnicas para obtención del foco de la pregunta para detectar si algún patrón supone mayor reto para los modelos.

Al respecto de tipos de preguntas por su palabra de comienzo se han considerado gurpos que al menos consten de 15 preguntas en entrenamiento o 7 preguntas para la partición de desarrollo para su análisis. En la figura 3.5 se tiene que las preguntas que comienzan con verbos (*are*, *can*, *did*, *do*, *does*, *is*, *was*) tienen un porcentaje de error en entrenamiento mayor que el resto de palabras, entre el $26\,\% - 38\,\%$. Las partículas interrogativas registran los siguientes errores: *why* un $26\,\%$ en entrenamiento y $56\,\%$ en *dev*, *what* un $13\,\%$ en entrenamiento y $26\,\%$ en desarrollo, *where* un $12\,\%$ y $24\,\%$, *how* un $11\,\%$ y $23\,\%$.

Las caracterizaciones por trigramas morfosintácticos para grupos de al menos 10 preguntas (disponible detalle en la figura 3.6) permite observar que las preguntas donde se localizan más errores cumplen el trigrama *WP VBZ CD*: $35\,\%$ en *train*, $49\,\%$ en desarrollo. Un ejemplo de pregunta de este grupo es: *What is one of the oldest uses of solar energy?*

Figura 3.5: *Relación entre el tipo de partícula interrogativa y errores cometidos en SQuAD v2.0 train y dev*

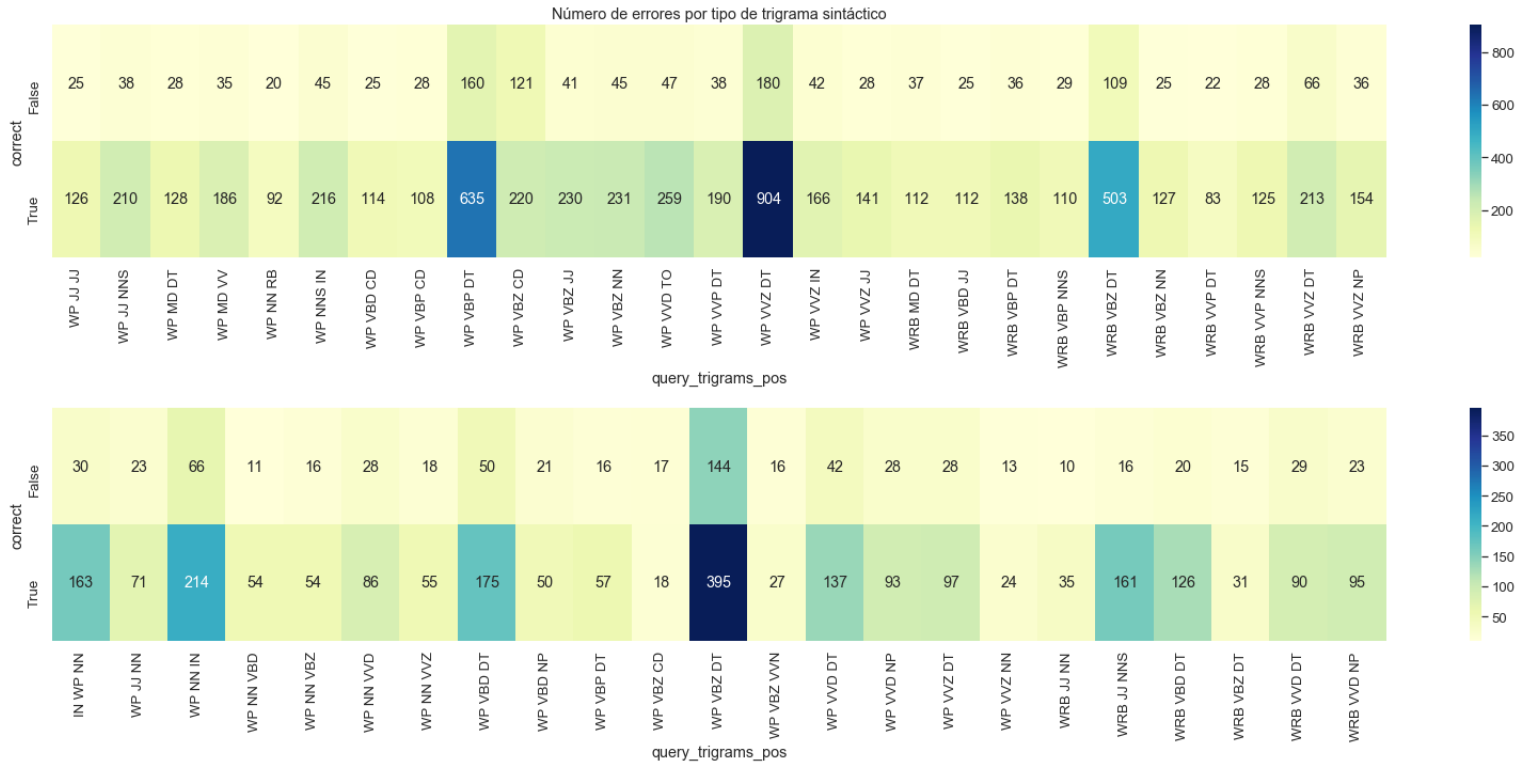


Figura 3.6: *Relación entre el tipo de trigrama morfosintáctico y errores cometidos en SQuAD v2.0 train y dev*

En los focos con un porcentaje de error elevado, con un mínimo de 7 preguntas por grupo (figura 3.7), se detectan palabras poco concretas al respecto del tipo de información de respuesta, por lo que denota más exigencia en cuanto a análisis e inferencia al contestar (se incluye porcentaje de error en entrenamiento): *difference* (30 %) y *differences* (36 %), *way* (32 %), *object* (40 %), *someone* (43 %), *something* (47 %), *thing* (50 %), *view* (55 %). En desarrollo destaca el foco *reason* con 85 % de error y el bajo error del foco *person* (19 %) y *time* (12 %).

Figura 3.7: *Relación entre el tipo de foco de la pregunta y errores cometidos en SQuAD v2.0 train y dev*

3.3.3 Análisis de errores por tipos de respuestas

El análisis de errores de los modelos por grupos de respuestas permite detectar si por su caracterización lingüística supone más dificultad para la extracción del fragmento, tarea que realizan los modelos al seleccionar sintagmas de distintos tipos o fragmentos que contengan entidades reconocidas.

En la figura 3.8 se muestra un mapa de calor relacionando el número de preguntas respondidas correcta e incorrectamente con el tipo de respuesta esperado. Se observa que se tiene mayor dificultad para las respuestas que suponen extraer un sintagma verbal ya que hay 21 % de error en entrenamiento, 32 % en desarrollo (por ejemplo *multiplying two integers*) y adjetivas detectando un 16 % de error en entrenamiento, 32 % en desarrollo (por ejemplo *nonconservtive forces*). En el caso de sintagmas verbales, se encuentra en una de las tipologías de tipos de respuesta menos frecuentes para la colección *SQuAD* como se puede ver en la tabla 3.7.

Por tipos de entidades reconocidas en las respuestas por grupos de al menos 15 respuestas en la colección de entrenamiento, se detectan confusiones con respuestas que contienen simultáneamente tanto numeros y fechas (20 %), localizaciones y países (15 %) y organizaciones y números (14 %). Para la colección de desarrollo se observan errores con entidades únicas en la respuesta: 35 % en miscelánea, 35 % en nacionalidades, 28 % en ordinales, 28 % en títulos de personas y en conjunto título y personas (23 %). Se puede ver más detalle en la figura 3.9.

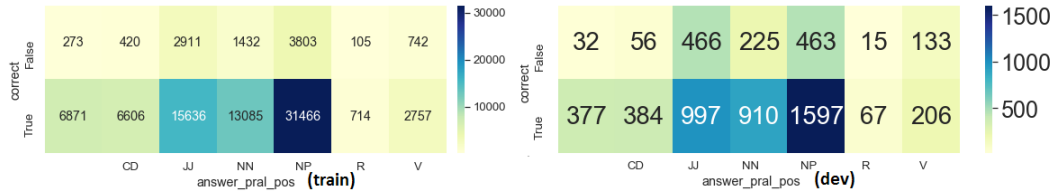


Figura 3.8: *Relación entre el tipo de respuesta sintáctica y errores cometidos en SQuAD v2.0 train y dev*

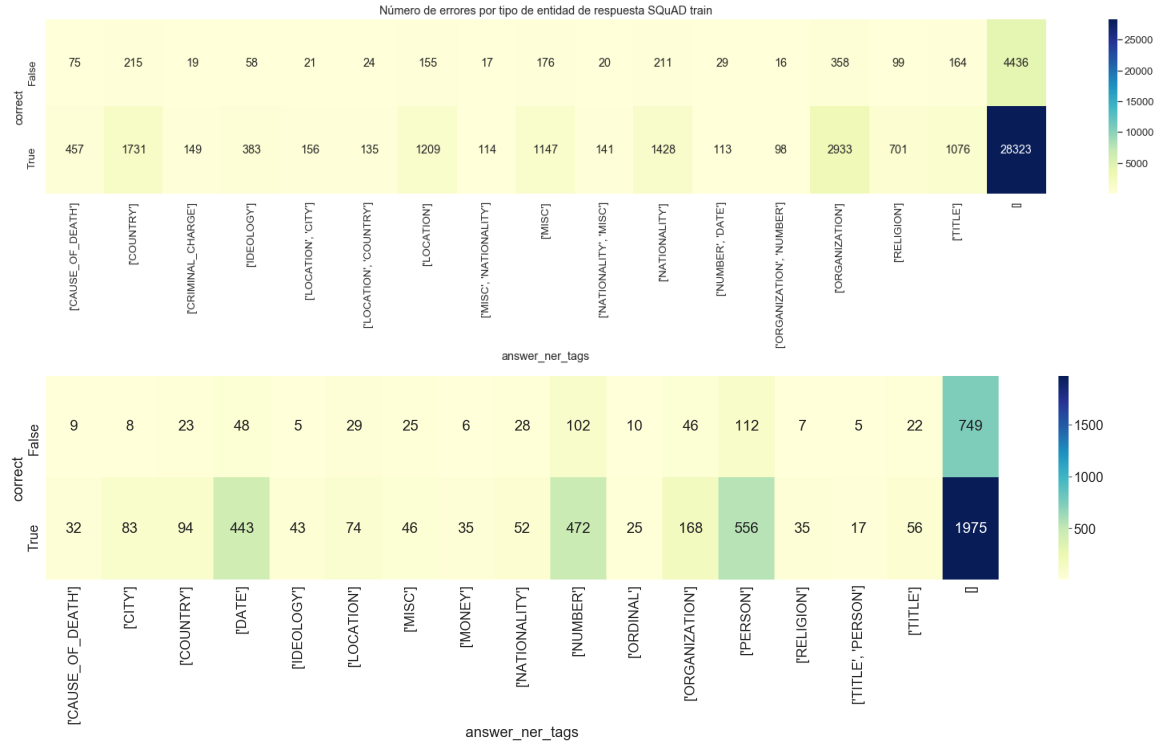


Figura 3.9: *Relación entre el tipo de respuesta por entidad reconocida y errores cometidos en SQuAD v2.0 train y dev*

3.3.4 Análisis de errores en clases relativas al foco de la pregunta y entidad de la respuesta

Para todos los modelos con entrenamiento sobre la colección objeto de análisis, se han obtenido los resultados sobre todas las preguntas falladas por alguno de los modelos en relación al tipo de foco. La figura 3.10 muestra número de ejemplos por asociación automática (si ha sido posible, sino se indica fallo como *KO*) del foco de la pregunta y la entidad contenida en la respuesta (si no hay entidad, no aplica y se indica *NA*).

Se observa que en la colección de *SQuAD* no es posible evaluar las clases asociadas a cantidades monetarias, porcentajes y duración entre otras: no se ha podido poblar estos grupos con suficientes ejemplos con la validación del foco de la pregunta y entidad de la respuesta automáticamente con las reglas de asociación entre el foco de la pregunta y la entidad detectada, además de ser entidades escasas en la colección (ver).

En la partición de entrenamiento de *SQuAD* se tiene un 18 % de error para preguntas que asocian foco y entidad del tipo localización, seguida de un 17 % para la clase de organizaciones (20 % de error en desarrollo). Se obtiene en general altos porcentajes de acierto para números (87 % de acierto en desarrollo) y tiempos (92 % de acierto en desarrollo).

Para evaluar resultados en particular para cada modelo, se muestra los resultados con más detalle en las tablas del anexo B.5 y B.4.

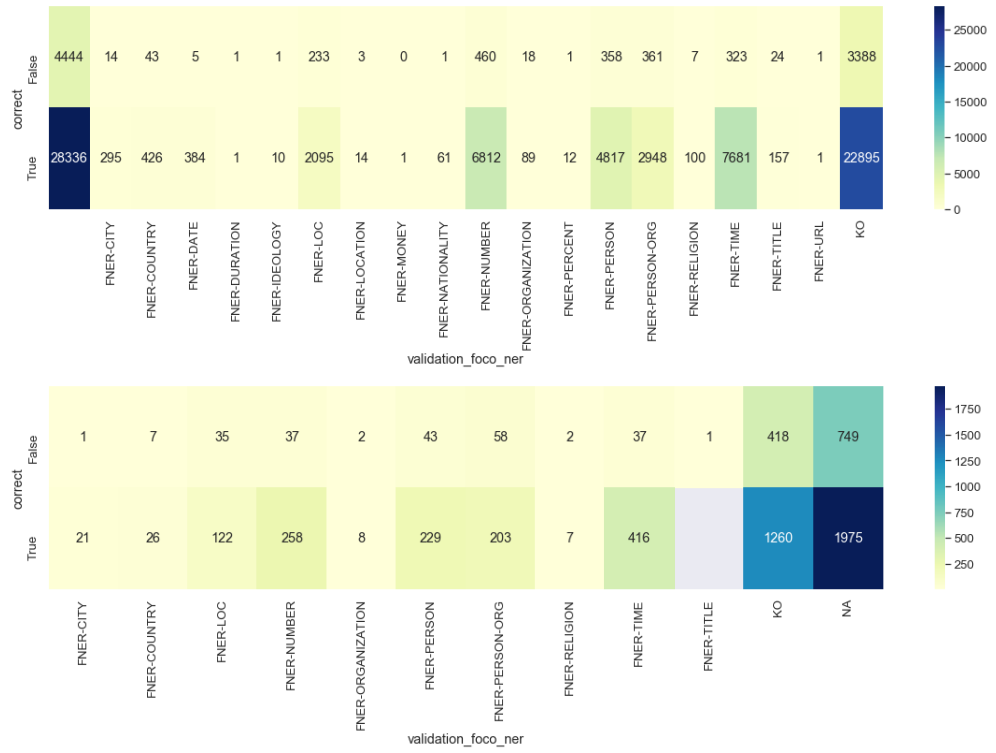


Figura 3.10: *Relación entre el foco de la pregunta y tipo de respuesta por entidad reconocida en SQuAD v2.0 train y dev*

3.3.5 Análisis de la confianza de respuesta

Los Modelos Neuronales de Lenguaje para Pregunta-Respuesta extractivo al dar respuesta aportan la confianza de predicción en su salida. En la figura 3.11 se representan las distribuciones de la variable aleatoria asociada a la confianza de cada modelo para las preguntas que tienen respuesta (preguntas *no null*) para las preguntas falladas por alguno de los modelos.

Para las preguntas contestables de la muestra con errores la confianza

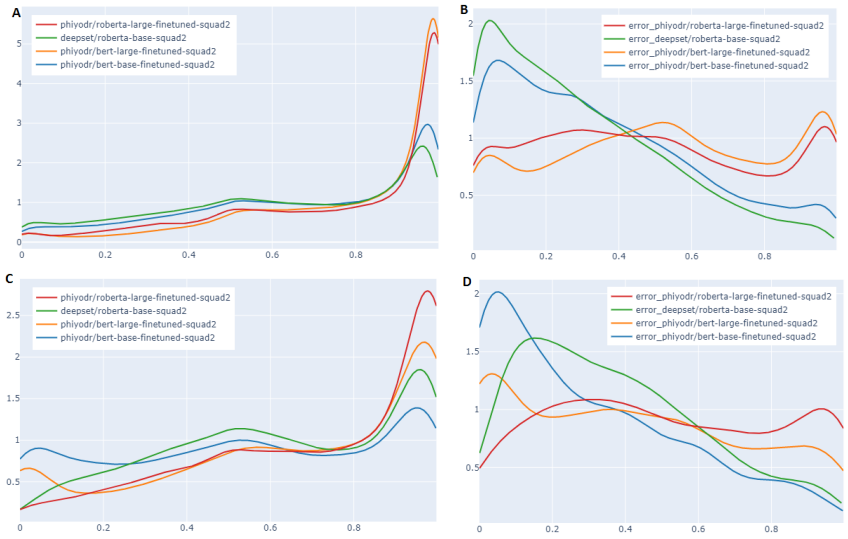


Figura 3.11: *Distribución de confianza de respuesta (KDE) para SQuAD train (A) y dev (C) para BERT-base, BERT-large, RoBERTa-base y RoBERTa-large. Análogo para partición con errores de SQuAD train (B) y dev (D).*

que arrojan los modelos desciende, en el 75 % de los casos por debajo del 54 % (*train*) y 48 % (*dev*) de confianza para *BERT base* y por debajo del 47 % (*train*) y 54 % (*dev*) para *RoBERTa base* (cuartil 75 %). El 50 % de la distribución de *BERT large* es inferior al 53 % (*train*) y 39 % (*dev*) de confianza, inferior al 47 % (*train*) y 49 % (*dev*) para *RoBERTa large* (cuartil 50 %). Comparando los percentiles obtenidos en el caso de *BERT* en sus dos versiones se observa que en la colección de entrenamiento los percentiles superan el valor del mismo percentil de la partición de desarrollo, lo que denota que la colección de entrenamiento se emplea para tal uso y supone sobreajuste que no se dispone sobre la colección de desarrollo.

Con una visión conjunta de las distribuciones y los percentiles se observa que ante preguntas consideradas como erróneas por este procedimiento ingenuo (al menos algún modelo la ha fallado) puede ocurrir que la confianza de respuesta sí supere el 50 % en los modelos con mayor número de parámetros (versión *large*), dado que los percentiles 50 % son cercanos al 50 % tanto para *BERT* como para *RoBERTa*: tienen una distribución balanceada tanto en entrenamiento como en desarrollo; mientras que en el acierto sí es frecuente que aporten una métrica alta de confianza. Sin embargo, los modelos con menor número de parámetros, su versión *base*, sí responden con confianzas menores ante esta selección de preguntas anotadas como erróneas, como se observa en sus percentiles del 75 %; su comportamiento ante el acierto es más moderado en cuanto a confianzas altas que las versiones *large*.

3.4 Análisis de dificultad por caracterización de las colecciones

A partir de los resultados obtenidos con las técnicas de Procesamiento de Lenguaje Natural para análisis sintáctico, reconocimiento de entidades y obtención de foco de la pregunta, es posible comparar las colecciones por sus etiquetas lingüísticas para cada partición para sus preguntas, respuesta y contexto. En concreto, en esta sección se ha comparado las colecciones de planteamiento extractivo *SQuAD* (sobre Wikipedia) y *NewsQA* (sobre noticias de *CNN*) del planteamiento de opción múltiple de *RACE*.

Por el origen de la colección se observa que la longitud del contexto a tratar es mayor para *NewsQA* y para la colección compleja de *RACE*, *RACE-H* (ver tabla 3.1). Las preguntas son similares por su longitud en *SQuAD* y *RACE-H* y similares en cuanto a partícula interrogativa (*what*) pero se distinguen por el uso de *who* más común en las dos colecciones de pregunta-respuesta extractivo mientras que *which* destaca en *RACE* (ver tabla 3.2). La estructura sintáctica por sus trigramas morfosintácticos de comienzo diferencial de *RACE* consta de sintagmas verbales para denotar opinión o referencia a inferir en del texto *VVG TO DT* (*According to the...*) al igual que sus focos frecuentes *following, writer, purpose, statements*, mientras que *SQuAD* propone más preguntas con sintagmas preposicionales *IN WP NN* (por ejemplo *In what country...*) y *NewsQA* preguntas con referencias a entidad nombrada por tener un sustantivo propio, por ejemplo *WP VVZ NP* como se observa en la tabla 3.4.

Otra clasificación de las preguntas realizada es por el tipo de respuesta esperada. En el caso de *SQuAD* y *NewsQA* denotan que en Pregunta-Respuesta extractivo es habitual el tipo de preguntas factuales o de hechos concretos, ya que se entrena los modelos con al menos un 35 % de respuestas con entidad nombrada (ver tabla 3.8), a diferencia de las respuestas de resumen u opinión del planteamiento de opción múltiple de *RACE*. Además, tanto en *SQuAD* como en *NewsQA* la entidad más solicitada es persona u organización: foco *person* solicitada de forma explícita como muestra la clase frecuente *FNER-PERSON* que relaciona con el foco con respecto a la clase

ambigua *FNER-PERSON-ORG*). A continuación, las respuestas numéricas y temporales (fechas). Sin embargo, en ambas es poco frecuente solicitar cantidades monetarias, porcentajes, catástrofes o causas de muerte y crímenes, entre otras; pero sí aparecen múltiples entidades en la respuesta, entre un 3 % y 7 % de los casos, más habitual en *NewsQA*. En cuanto a la formulación de las respuestas es habitual detectar sustantivos propios asociados a las entidades, seguido de sintagmas adjetivales o sustantivas; más sintagmas verbales en *NewsQA*.

La asociación automática entre el foco de la pregunta y entidad a extraer como respuesta da como resultado particiones de preguntas reducidas lo que denota que el foco no es explícito ni aporta pista para la extracción de la respuesta (casos no *null*), ya que solo el 30 % de las respuestas han podido vincularse al foco de manera automática, tanto en *SQuAD* como *NewsQA* (ver tabla 3.10). Esto implica que se requiere de más conocimiento general y semántica no enriquecida en el método automático (sinonimia, hiperonimia, hiponimia, etc.) para identificar el tipo de entidad, denotando mayor variabilidad lingüística *NewsQA* que *SQuAD* en otras entidades de respuesta no vinculadas por reglas de asociación.

Respecto a los errores cometidos por los modelos entrenados con *SQuAD* sobre dicha colección por caracterización lingüística, se observa que es más frecuente dar una respuesta errónea con cualquier modelo si se espera una respuesta de más de 11 palabras (ver figura 3.2). También, las preguntas que registran mayor número de errores implican a preguntas interrogativas poco

frecuentes en la colección como *why*, *where* y *how* o con focos poco concretos del tipo *view*, *thing*, *something*, *someone* y *object* (ver tabla 3.7). Análogamente, se responde peor a aquellos tipos de respuestas poco frecuentes como los sintagmas verbales (ver tabla 3.7).

En general, los patrones de errores generados por los cinco Modelos Neuronales de Lenguaje con adaptación sobre *SQuAD* evaluados en este trabajo (ver tabla 3.12) no destacan una partición o tipología con más errores acumulados de forma clara. Esto motiva sistemas de predicción de dificultad automática con técnicas de Aprendizaje Automático que en base a la caracterización lingüística modelada en sus variables den como salida la dificultad para esa pregunta, contexto y modelo, dificultad general o detallada por niveles.

Capítulo 4

Metodología para la anotación automática de dificultad

Como se ha observado en el estudio del estado del arte, a partir de un texto determinar si una pregunta y respuesta es difícil mediante un juicio único pactado para una anotación homogénea es una tarea que requiere gran esfuerzo manual de anotación, por lo que se realiza sobre un número reducido de ejemplos de las colecciones. Además, en el estudio del estado del arte se observa diversificación en cuanto al dominio y forma de creación de las colecciones y fenómenos lingüísticos presentes en ellas para determinar dificultad.

El proceso de anotación automático propuesto en este trabajo se basa en los errores cometidos por Modelos Neuronales de Lenguaje, contemplando

un modelo en concreto o por los errores cometidos por la mayoría de ellos, dado que es común acertar o fallar las mismas preguntas.

4.1 Criterios de comparación entre respuestas

Para realizar la anotación automática de la colección por su dificultad, es necesario considerar un criterio de comparación para cada respuesta dada por un modelo de forma que se juzgue como erróneo o no con él. El criterio más básico consiste en la comparación literal palabra a palabra de la respuesta dada por el modelo con respecto a la respuesta *gold standard* o de refencia de la colección.

En este trabajo se realiza un preprocesado, retirado de partículas sin significado y normalización de símbolos de puntuación en ambos textos de respuesta previamente a su comparación. También se valora la respuesta como correcta si la respuesta aportada está contenida en ella, es decir, las palabras coinciden literalmente de forma consecutiva (*word matching*). Para más detalle ver el desarrollo disponible en el anexo A.7.

En base a este criterio de comparación, en este trabajo se proponen 3 modelos distintos de dificultad, dos de ellos binario y uno con tres clases para reflejar distintos niveles de dificultad. Para este planteamiento de anotación, se consideran las preguntas con respuesta (preguntas no *null*) para el entrenamiento del modelo predictor de dificultad.

Queda abierta la propuesta de un modelo para las preguntas sin respuesta y para ello su anotación por dificultad para preguntas no contestables, por ejemplo, a partir del criterio comparación mediante confianzas de respuesta o comparación de otras respuestas posibles aportadas en la colección (*plausible*) con las respuestas dadas por los Modelos Neuronales de Lenguaje.

.

4.2 Anotación por errores binaria (binario ingenuo)

En este modo de anotación, una pregunta se considera de una de las dos formas antónimas: fácil o difícil. En el caso de que al menos un sistema haya fallado la pregunta, esta se anota como difícil. Si todos los sistemas la aciertan, entonces se anota como fácil.

Este criterio se ha determinado para considerar todas las posibles dificultades a partir de los errores de cada modelo en particular, de forma que al aplicar el modelo predictor entrenado con esta colección sobre distintos Modelos Neuronales de Lenguaje permite ver robustez entre ellos, es decir, si los errores cometidos entre ellos tienen similitudes en cuanto a la caracterización lingüística reflejada transmitida en el ejercicio de anotación. Es decir, este criterio de dificultad sobreajusta a dificultades de cada modelo para ver su comportamiento ante similitudes presentes en distintas colecciones de Pregunta-Respuesta por su caracterización, en ellas también es esperable el mismo patrón de dificultad.

4.3 Anotación por dificultad binaria por mayoría

La anotación de dificultad por errores cometidos para algún modelo de Búsqueda de Respuestas es una estrategia generosa a la hora de considerar preguntas difíciles para los modelos en base a todas las falladas. En el análisis de errores, se ha observado que los Modelos Neuronales de Lenguaje tienen planteamientos y arquitecturas similares y en este trabajo se corrobora que cometen frecuentemente los fallos en las mismas preguntas (ver sección 3.3).

Por ello, en este planteamiento se propone una modificación en la que se considera difícil una pregunta si más de la mitad de los sistemas de Pregunta-Respuesta evaluados fallan, dando lugar a una dificultad binaria por mayoría. Este plantamiento no tiene como objetivo analizar el resultado en particular de un solo modelo neuronal de Búsqueda de Respuestas sino las dificultades comunes de todos los modelos.

4.4 Anotación por dificultad multiclase

Este método de anotación se propone para un modelo predictor que tena en cuenta tres niveles de dificultad: fácil, media o alta. Una pregunta de dificultad media se considera a aquella que ha sido respondida parcialmente, es decir, parte de las palabras en la respuesta son aceptadas por una persona ya que la respuesta aportada está contenida en la correcta o viceversa, es decir, hay selección errónea al comienzo o en la finalización del fragmento

extraído como respuesta por el Modelo Neuronal de Lenguaje.

Para ello, caracterizar si una pregunta tiene dificultad baja (fácil), media o alta (difícil) se propone una anotación en base a los errores cometidos por los Modelos Neuronales de Lenguaje con entrenamiento en una colección específica y observando coincidencias (términos comunes consecutivos) entre las respuestas dadas por los modelos y la respuesta *gold standard*.

Para considerar una pregunta como fácil la mayoría de los modelos deben acertarla. Análogo para las preguntas difíciles, siguiendo el planteamiento de dificultad por mayoría. Adicionalmente, si una pregunta no es acertada ni fallada por la mayoría y se acierta parcialmente por algún modelo se considera de dificultad media.

Capítulo 5

Resultados de la anotación automática de dificultad

En este trabajo se han seleccionado las siguientes colecciones de Pregunta-Respuesta extractivo: *SQuAD* con un total de 92.749 preguntas contestables y *NewsQA* con 55.961 preguntas contestables.

En el caso de *SQuAD 2.0* se han considerado 86.821 (*train*) y 5.928 (*dev*) preguntas y respuestas asociadas a los textos al omitir aquellas preguntas que por definición no son contestables. Para su anotación automática por dificultad se emplea los modelos entrenados con *SQuAD* considerados en la tabla 3.11 y los errores cometidos.

De forma análoga, para *NewsQA* en sus particiones de entrenamiento y evaluación se tiene 57.210 y 3.218 preguntas contestables, respectivamente,

anotadas por dificultad con los modelos entrenados con *NewsQA* considerados en la tabla 5.1 en base a los errores cometidos con la comparación automática entre respuestas.

Para cada grupo de modelos con adaptación en una colección concreta, se anota la partición de entrenamiento y una de las disponibles para evaluación (*dev* o *test*) con sus errores, ya que serán las empleadas para el entrenamiento y evaluación del modelo predictor de dificultad. Así, para los modelos entrenados con *SQuAD* se anota la partición de entrenamiento y su versión de *dev* y, sin embargo, la colección de *test* de *NewsQA*. Y viceversa para los modelos con entrenamiento en *NewsQA*, se anota la colección de *dev* de *SQuAD* para observar variación en la distribución de preguntas anotadas como difíciles.

Se plantea esta metodología ya que al anotar simultáneamente tanto *dev* y *test* para una misma colección es esperable obtener una distribución similar de preguntas anotadas por dificultad y no se va a emplear ambas para la evaluación del modelo predictor por estar igualmente distribuídas. Emplear la distribución de anotación en una nueva colección sobre la que no se ha entrenado permite comparar la distribución de preguntas anotadas como difíciles en una nueva colección sobre la que los Modelos Neuronales de Lenguaje no han sido entrenados.

A continuación se expone los resultados obtenidos para cada método de anotación. En todos ellos se observa que los modelos entrenados sobre

Modelo	Fine-tunning	Modelos
BERT	*NewsQA*	tli8hf/unqover-bert-base-uncased-newsqa tli8hf/unqover-bert-large-uncased-newsqa
RoBERTa	*NewsQA*	tli8hf/unqover-roberta-base-newsqa tli8hf/unqover-roberta-large-newsqa

Tabla 5.1: *Modelos de HuggingFace para Pregunta-Respuesta con fine-tunning en NewsQA*

NewsQA cometen más errores sobre las particiones de *SQuAD dev* y *NewsQA test*, por lo que la distribución de preguntas difíciles aumenta para todos los planteamientos de anotación con modelos entrenados con *NewsQA*.

5.1 Anotación binaria ingenua

En esta anotación, como se ha descrito en la sección 4.2, se anota como difíciles aquellas preguntas en las que al menos un Modelo Neuronal de Lenguaje ha arrojado una respuesta incorrecta.

Para ello se emplean modelos adaptados en las colecciones de Búsqueda de Respuestas, contempladas sus versiones base o en mayor número de parámetros y especializados en *SQuAD* (ver tabla 3.11) y *NewsQA*[1] (ver tabla 5.1) con su nombre y los resultados reportados oficialmente[2].

Así se obtiene las anotaciones con la distribución de la tabla 5.4 teniendo

[1]No se dispone de modelos para Pregunta-Respuesta públicos *XLNet* y *T5* para *NewsQA*

[2]En el caso de *NewsQA* no se suministran por no haber resultados publicados.

en cuenta los resultados obtenidos con los cinco Modelos Neuronales de Lenguaje con *fine-tunning* sobre *SQuAD* (ver 3.11) y para los cuatro modelos con especialización en *NewsQA*, la tabla 5.5 (ver 5.1).

Colección	Dificultad	train	dev	test
SQuAD 2.0	Fácil	77.135	4.538	-
	Difícil	9.686	**1.390**	-
NewsQA	Fácil	-	-	1.211
	Difícil	-	-	**2.007**

Tabla 5.2: *Anotación binaria ingenua con Modelos Neuronales de Lenguaje con fine-tunning sobre SQuAD preguntas no null*

Se observa que para la colección *NewsQA* predominan las preguntas difíciles, mientras que en *SQuAD* para los modelos entrenados con ella se anotan menor proporción de casos como difíciles, al contrario que para los modelos entrenados con *NewsQA* al aplicar la anotación sobre la colección *SQuAD dev*.

Colección	Dificultad	train	dev	test
SQuAD 2.0	Fácil	-	2.419	-
	Difícil	-	**3.509**	-
NewsQA	Fácil	20.320	-	136
	Difícil	36.890	-	**3.082**

Tabla 5.3: *Anotación binaria ingenua con Modelos Neuronales de Lenguaje con fine-tunning sobre NewsQA sobre preguntas no null*

5.2 Anotación binaria por mayoría

Para esta anotación de dificultad se tiene en cuenta si al dar respuesta la mitad o la mayoría de los Modelos Neuronales de Lenguaje han fallado la pregunta para considerarla como difícil, como se ha descrito en 4.3. Por tanto, este método es más exigente en cuanto a la consideración de dificultad por lo que se observa descenso en el número de preguntas anotadas como difíciles.

Se obtiene las anotaciones con la distribución de la tabla 5.4 teniendo en cuenta los resultados obtenidos con los cinco Modelos Neuronales de Lenguaje con *fine-tunning* sobre *SQuAD* (ver 3.11). En los resultados obtenidos sobre *SQuAD* predominan los casos fáciles, mientras que en el caso de *NewsQA* se tiene una anotación más equilibrada de ejemplos fáciles y difíciles, también en *train*.

Colección	Dificultad	train	dev	test
SQuAD 2.0	Fácil	84.802	5.445	-
	Difícil	2.019	**483**	-
NewsQA	Fácil	-	-	1.826
	Difícil	-	-	**1.392**

Tabla 5.4: *Anotación binaria por mayoría con Modelos Neuronales de Lenguaje con fine-tunning sobre SQuAD preguntas no null*

Para los cuatro modelos con especialización en *NewsQA* se tiene la distribución de la tabla 5.5 (ver 5.1).

Colección	Dificultad	train	dev	test
SQuAD 2.0	Fácil	-	4.852	-
	Difícil	-	**1.076**	-
NewsQA	Fácil	33.582	-	1.618
	Difícil	23.628	-	**1.600**

Tabla 5.5: *Anotación binaria por mayoría con Modelos Neuronales de Lenguaje con fine-tunning sobre NewsQA sobre preguntas no null*

5.3 Anotación por dificultad multiclase

Como se ha descrito en la sección 4.4, adicionalmente a la anotación binaria por mayoría, si una respuesta es parcialmente correcta debido a una selección errónea del fragmento de respuesta del texto o si no ha sido fallada por la mayoría se considera de dificultad media.

En la tabla 5.6, se muestran las distribuciones de ejemplos anotados teniendo en cuenta por una parte los resultados de los cinco Modelos Neuronales de Lenguaje con *fine-tunning* sobre *SQuAD* (ver 3.11) y por otra los modelos con especialización en *NewsQA*, la distribución de la tabla 5.7 (ver 5.1).

Se observa que las preguntas disponibles en *train* sobre *SQuAD* predomina de nuevo las fáciles, mientras que en *NewsQA train* predomina las preguntas difíciles. Respecto a los resultados de preguntas medias, se tiene coincidencia en *SQuAD train* y *NewsQA train* para los modelos entrenados

con ellas, respectivamente, en cuanto al número de preguntas consideradas como medias, un 20 % de los casos se emplearán para entrenamiento.

Colección	Dificultad	train	dev	test
SQuAD 2.0	Fácil	67.055	3.823	-
	Media	17.467	1.622	-
	Difícil	2.299	**483**	-
NewsQA	Fácil	-	-	826
	Media	-	-	930
	Difícil	-	-	**1.462**

Tabla 5.6: *Anotación multiclase con Modelos Neuronales de Lenguaje con fine-tunning sobre SQuAD preguntas no null*

Colección	Dificultad	train	dev	test
SQuAD 2.0	Fácil	-	2.227	-
	Media	-	2.455	-
	Difícil	-	**1.246**	-
NewsQA	Fácil	20.903	-	916
	Media	11.782	-	588
	Difícil	24.525	-	**1.714**

Tabla 5.7: *Anotación multiclase con Modelos Neuronales de Lenguaje con fine-tunning sobre NewsQA sobre preguntas no null*

Capítulo 6

Predicción de dificultad

Una vez se ha propuesto una metodología automática de dificultad y tras haberse aplicado en distintas colecciones, en este capítulo se prueba si se puede realizar una predicción de forma automática por caracterización de la colección. Dada una colección de Pregunta-Respuesta se emplean las particiones de entrenamiento, desarrollo y evaluación (*test*) por separado para el entrenamiento del modelo predictor de dificultad (con ejemplos de la partición de entrenamiento) y se evalúa con una partición de la misma colección: bien la partición de *test* o bien la partición de desarrollo si no es pública la de *test*. Para cada una se ha considerado las preguntas contestables por suponer mayor reto para obtener un modelo predictor de dificultad a partir de las variables obtenidas por la caracterización lingüística y formulación del lenguaje tanto de la pregunta, respuesta a extraer y texto del que provienen.

Para ello, se ha generado variables en base a formulación de la pregunta y caracterizaciones lingüísticas del contexto y respuesta obtenidas con técnicas de análisis sintáctico, detección de entidades nombradas, obtención del foco de la pregunta y más técnicas de transformación, normalización y filtrado tras el estudio de importancia de las variables para el modelo de dificultad.

Teniendo en cuenta que los Modelos de Neuronales de Lenguaje para Búsqueda de Respuestas están especializados en una colección concreta y son evaluados sobre una partición con la misma distribución en cuanto a caracterización lingüística, el modelo predictor de dificultad puede quedar sobreajustado en su aprendizaje en base a los errores cometidos por los modelos sobre dicha coleccion en los que se especializaron. Por ello, en cada sección se incluye también una evaluación adicional para ver capacidad de generalización de los resultados de predicción de dificultad sobre otra colección de Pregunta-Respuesta distinta, para lo que se emplea su partición de evaluación o desarrollo en su defecto.

En este trabajo, se realiza la experimentación sobre la colección *SQuAD* y la evaluación adicional sobre *NewsQA*. A continuación se incluye la evaluación de cada modelo predictivo diseñado en forma de *baseline* para cada estrategia y tipos de modelos empleados de Aprendizaje Automático supervisado.

6.1 Variables de dificultad por caracterización lingüística

Las etiquetas lingüísticas obtenidas a partir de técnicas de Procesamiento de Lenguaje para caracterización de la formulación de la pregunta, contexto y respuesta de entrada al modelo, permiten formar variables de entrada para el modelo de predicción de dificultad. Para cada tripleta de una colección de Pregunta-Respuesta, se preprocesa la pregunta, contexto y respuesta, y se aplica las mismas transformaciones descritas a continuación para obtener un formato válido para el modelo predictor de dificultad.

En primer lugar, se realiza agregados por suma agrupando variables relacionadas para reducir el número de etiquetas a variables de entrada del modelo de predicción de dificultad (ver la tabla 6.1). Es decir, sobre más de 80 etiquetas lingüísticas y caracterizaciones iniciales se obtiene un máximo de 24 variables. También, se realizan transformaciones de normalización y estandarización de valores mínimo y máximo. En el caso de variables categóricas, se asigna una etiqueta numérica a los elementos más frecuentes enumerados para acotar la representación y variedad de la entrada empleando *label encoders* basados en *one hot encoding*.

- Número de palabras de la respuesta, número de palabras de la pregunta y número de palabras del contexto.

- Número de palabras comunes entre la pregunta y el contexto. Este indi-

cador mide el fenómeno lingüístico de contextualización de la pregunta por términos comunes. En caso de pocos términos comunes, puede ser causa de variaciones ortográficas, léxicas y semánticas (sinonimia, antonimia, hiperonimia, hiponimia o paráfrasis).

- Número de oraciones en el contexto. Este indicador representa información para los casos en los que haya que procesar múltiples oraciones para extraer la respuesta.

- Tipo de término interrogativo (variable categórica): *what*, *who*, *when*, *where*, *which*, *how*, *why*, *in*, *the* u otro (*other*).

- Tipo de respuesta por análisis sintáctico (variable categórica): *NP*, *NN*, *JJ*, *V*, *R*, *CD* u otro (*other*).

- Tipo de foco de la pregunta (variable categórica): *person*, *organization*, *location*, *time*, *number*, sustantivo común (*NN*), sustantivo propio (*NP*), otro (*other*).

- Tipo de entidad o entidades múltiples de la respuesta (variable categórica): *PERSON*, *ORGANIZATION*, *NUMBER*, *PERCENT*, *MONEY*, *LOCATION*, *CITY*, *COUNTRY*, *STATE OR PROVINCE*, *TIME*, *DATE*, *DURATION*, *RELIGION*, *IDEOLOGY*, *NATIONALITY*, *ORDINAL*, *SET*, *TITLE*, *CAUSE OF DEATH*, *CRIMINAL CHARGE*, otro (*MISC*), múltiple (*MULTI*) o ninguno (*NONE*).

- Indicador de si se cumple o no la validación automática entre el foco de la pregunta y tipo de respuesta por su entidad.

- Número total de entidades por tipo detectadas en la pregunta y en el contexto agregadas: organizaciones, personas, título de personas, ideología, religiones, ciudades, países, localizaciones, nacionalidades, estados y provincias, fechas, duraciones, referencias temporales, porcentajes, números, ordinales, cantidades monetarias, causas de muerte y actos criminales. Este indicador mide dificultad por cambios de entidades o dificultad de selección de la entidad correcta entre varias del mismo tipo. Es sensible a preprocesados erróneos de lenguaje en la tarea de detección de entidades nombradas.

- Número total de etiquetas sintácticas por tipo obtenidas en el preprocesado de la pregunta y el contexto. Es decir, número de sustantivos comunes en singular (NN), número de sustantivos comunes en plural (NNS), adjetivos (JJ, JJR, JJS), verbos (VV, VVD, VVG, VVN, VVP, VVZ), conjunciones coordinadas (CC), cardinales (CD) adverbios Wh de lugar y tiempo (WRB), pronombres *who, what* (WP) y determinante *which* (WDT). Es sensible a preprocesados erróneos de lenguaje en la tarea de análisis sintáctico.

- Confianza arrojada por un Modelo Neuronal de Lenguaje de Búsqueda de Respuestas al contestar.

A continuación se expone el estudio de la importancia de las variables con los ejemplos de la partición de entrenamiento de la colección de referencia y en base a los errores contemplados en el análisis por caracterización lingüís-

Variable	Detalle
len_ answer	Longitud de la respuesta
len_ query	Longitud de la pregunta
len_ context	Longitud del contexto
num_ common_ terms	Número de términos comunes pregunta y contexto
num_ context_ sentences	Número de oraciones en el contexto
model_ confidence	Confianza del modelo al dar respuesta
wh_ query	Término interrogativo: *what*, *who*, *where*, etc.
ner_ type_ answer	Tipo de entidad de la respuesta
answer_ pral_ pos	Tipo de respuesta por su análisis sintáctico
foco	Foco del tipo: NN, NNS, person, time, other, place
validation_ foco_ ner	Si se cumple o no validación automática foco entidad
NP_ sum	Suma de sustantivos pregunta y contexto: NP, NPS
NN_ sum	Suma de sustantivos pregunta y contexto: NN, NNS
JJ_ sum	Suma de adjetivos pregunta y contexto: JJ, JJR, JJS
VV_ sum	Suma de verbos pregunta y contexto: VV, VVD, VVG...
RB_ sum	Suma de adverbios pregunta y contexto: RB, RBR, RBS
PP_ sum	Suma de pronombres pregunta y contexto: PP, PP$
WH_ sum	Suma de partículas pregunta y contexto: *WDT*, *WP*, *WRB*
DT_ sum	Suma de determinantes pregunta y contexto: *DT*, *IN*, *CC*...
numerical_ references	Suma de PERCENT, NUMBER, ORDINAL, MONEY y números (*CD*, *LS*)
place_ references	Suma de CITY, COUNTRY, LOCATION, NATIONALITY y STATE_OR_PROVINCE
person_ org_ references	Suma de PERSON y ORGANIZATION
temporal_ references	Suma de DATE, DURATION y TIME
other_ references	Suma de TITLE, IDEOLOGY, CASE_OF_DEATH, CRIMINAL_CHARGE, RELIGION

Tabla 6.1: *Variables finales de los modelos de predicción de dificultad*

tica para preguntas contestables (sección 3.3) con respecto a no contestables.

6.1.1 Importancia de las variables por resultados obtenidos con los modelos

Como estudio previo de las variables se observa correlaciones entre ellas. Cuantas más palabras tiene un texto, mayor número de sustantivos, adjetivos y más variables de suma de tipos por etiquetas sintácticas se anotan, estando correlacionadas de forma positiva (azules más oscuros y marcados en la figura 6.1).

Sin embargo, se observa correlación negativa entre el tipo sintáctico de respuesta con otras variables: confianza de la respuesta del modelo neuronal, el número de palabras de la respuesta y la validación entre el foco de la pregunta y de la respuesta.

Para determinar la importancia de las variables en la representación por errores cometidos se ha empleado un modelo previo para selección de variables basado en árboles de decisión y clasificación: *ExtraTreesClassifier*.

En función de si en la predicción de dificultad se añade la tarea de considerar dificultad en preguntas no contestables (*null*) se observa variación en la importancia de las variables relacionada con las correlaciones negativas de la figura 6.1.

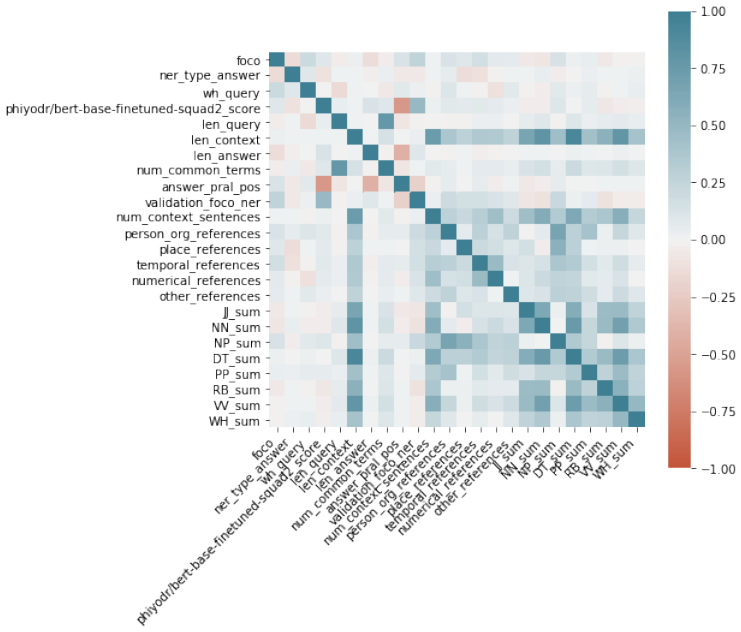


Figura 6.1: *Correlación de las variables por caracterización lingüística para la colección SQuAD train*

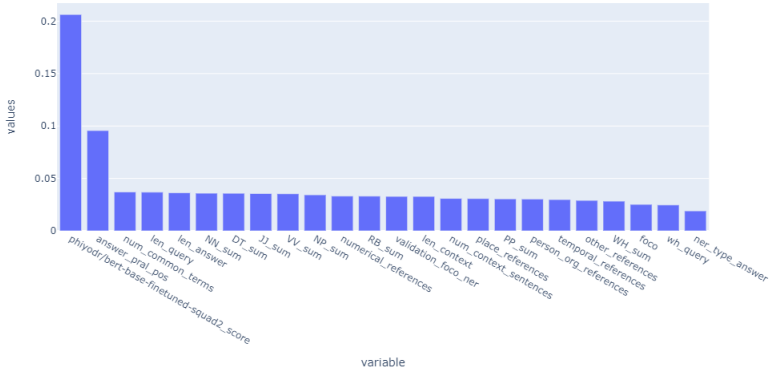


Figura 6.2: *Importancia de las variables por caracterización lingüística para la colección SQuAD train*

Al analizar todas las preguntas de la colección (incluídas *null*) en la figura 6.2 se observa que las variables con mayor importancia para la colección de referencia en el análisis de errores (*SQuAD*): el valor de confianza de respuesta del Modelo Neuronal de Lenguaje, el tipo sintáctico de la respuesta (en caso de vacía se incluye como *other*), número de términos comunes entre pregunta y contexto, la longitud de la pregunta y la longitud de la respuesta a extraer.

Al añadir la respuesta correcta esperada e información sobre su tipo sintáctico y longitud, los valores otros, cero y confianza de respuesta menor aporta información que permite detectar de forma directa preguntas no contestables. Las respuestas de los modelos sobre estas preguntas se comparan siguiendo el mismo criterio de comparación (ver el desarrollo A.7) con la segunda respuesta correcta posible (*plausible*) presente en las colecciones.

Considerando solo las preguntas *no null* o contestables, la variable sobre el tipo sintáctico de la respuesta pierde importancia siendo de las menos relevantes, como se puede ver en la figura 6.3. Además, en la figura 6.4 aparece representada información más detallada al respecto de correlación de las variables de más valor por formulación y caracterización del lenguaje como la influencia en la confianza de respuesta del modelo con mayor longitud de la respuesta a extraer y relación inversa entre el tipo de partícula interrogativa y la longitud de la pregunta, además de la relación entre el tipo de foco y el tipo de entidad asociada en la respuesta.

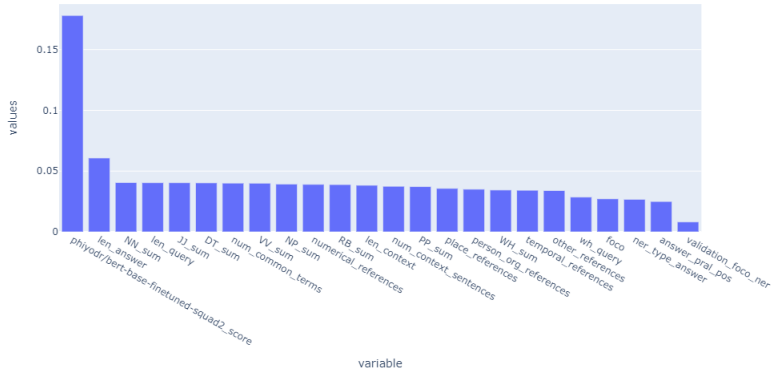


Figura 6.3: *Importancia de las variables por caracterización lingüística para la colección SQuAD train*

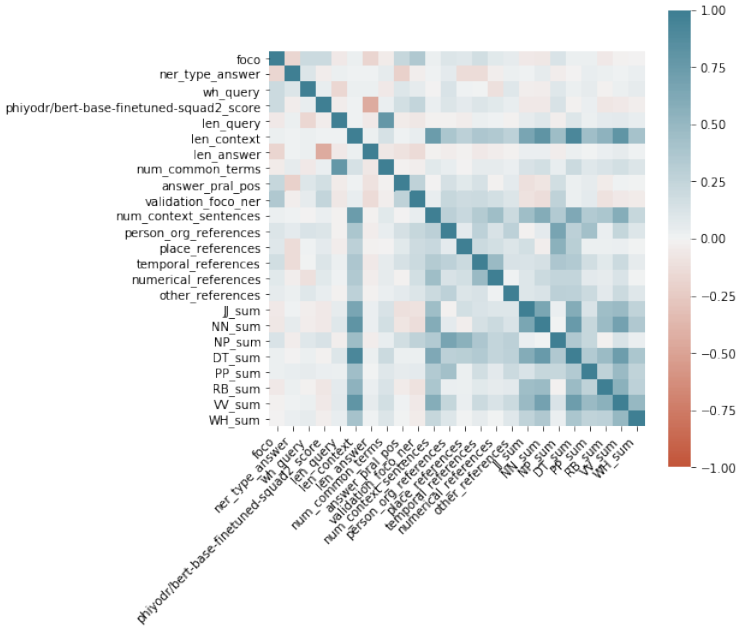


Figura 6.4: *Correlación de las variables por caracterización lingüística para las preguntas contestables de la colección SQuAD train*

Para la predicción de dificultad se valora más relevante la evaluación de los modelos ante preguntas contestables a partir de las variables consideradas y así obtener resultados a partir de los Modelos Neuronales de Lenguaje del estado del arte, ya que las técnicas para determinar si el modelo debe contestar (o no) no se refleja en las respuestas obtenidas con los modelos. Es decir, se requiere un procesado posterior en base a un ajuste de umbrales y la confianza del modelo para decidir *no contestar*, siendo una tarea a tratar de forma separada para considerar su dificultad y caracterización por la pregunta y el lenguaje del contexto y sin tener en cuenta la caracterización de las respuestas y sus variables asociadas.

6.2 Modelo de predicción de dificultad

El modelo predictivo o predictor de dificultad emplea una representación en forma de rasgos a partir de las variables obtenidas de la caracterización y formulación del lenguaje. Esta representación se forma para cada ejemplo de una colección de Pregunta-Respuesta a su entrada para componer el espacio vectorial a partir de rasgos o variables que caracterizan longitudes, tipologías, validaciones y totales encontrados (frecuencia normalizada) para cada texto de la tripleta de entrada (contexto, pregunta y respuesta de la colección de Pregunta-Respuesta).

Este planteamiento de modelo predictor de dificultad depende de la calidad y acierto de los modelos de análisis sintáctico y reconocimiento de

entidades empleados previamente, su robustez ante errores ortográficos y ambigüedades u omisiones del lenguaje dificultades propias previas de cada tarea. Las etiquetas lingüísticas quedan estandarizadas en forma y número a la entrada del modelo predictor de dificultad y puede asociar la dificultad a un modelo en particular al incluir la confianza de respuesta de un modelo de referencia como variable o predecir a partir de errores cometidos por una mayoría de los modelos de Pregunta-Respuesta para determinar la dificultad de la pregunta y respuesta a extraer sobre el texto de Comprensión Lectora. Al diseñar el modelo predictivo se ha seguido el planteamiento de comparativa o batalla de modelos para cada tipo de clasificación de dificultad, seleccionado adicionalmente diversas estrategias de Aprendizaje Automático supervisado: regresión, clasificación binaria y clasificación multiclase.

Las particiones de entrenamiento, desarrollo y evaluación (*test*) se emplean por separado de forma que una vez el modelo predictor de dificultad ha sido entrenado con la partición de entrenamiento se evalúa con una partición de la misma colección no contemplada previamente: la partición de *test* o bien la partición de desarrollo si no es pública la de *test*.

Generalmente en una misma colección la distribución de preguntas y caracterizaciones lingüísticas sobre los ejemplos es muy similar, por lo que adicionalmente se realizan pruebas del modelo predictor sobre otra colección de Pregunta-Respuesta que no ha visto en su entrenamiento y para modelos de Pregunta-Respuesta no especializados en ella, obteniendo los resultados de ambos modelos sobre la nueva colección no vista en procesos anteriores.

De este modo, se reproduce un entorno donde se dispone de un predictor entrenado que recibe una nueva colección a valorar y caraterizar por su dificultad.

El modelo predictor de dificultad puede diseñarse con distintas estrategias según los objetivos reflejados en la anotación de las colecciones de Búsqueda de Respuestas. A continuación, se propone diversos métodos de anotación para la predicción automática de dificultad en las colecciones.

6.3 Evaluación de dificultad con modelos con especialización sobre *SQuAD*

Para el entrenamiento de estos modelos predictores de dificultad se ha empleado la colección de *SQuAD* con su partición de entrenamiento sobre las preguntas contestables (86.821) y los Modelos Neuronales de Lenguaje con especialización en *SQuAD* (ver la tabla 3.11). Para sus primeras evaluaciones se han empleado 5.928 preguntas contestables de *SQuAD dev*. Para la evaluación adicional se emplea 3.218 preguntas contestables de *NewsQA test*.

6.3.1 Modelo de predicción de dificultad binario ingenuo

Para este planteamiento de modelo predictor se incluye la variable de confianza arrojada por un Modelo Neuronal de Lenguaje de referencia[1]. En la

[1]Se ha escogido la confianza de *BERT-base* por tener la distribución con valores inferiores en gran parte de sus respuestas con error y moderada en cuanto a valores altos de

sección 5.1 se ha descrito la distribución de preguntas sobre este criterio. Los resultados de predicción tras aplicar el modelo sobre esta anotación se muestran en las tablas 6.2 y 6.3.

Modelo *SQuAD train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.66	0.80	0.18	0.97	0.29	0.87
GaussianNB	0.50	0.83	0.41	0.88	0.45	0.85
SVC	0.00	0.77	0.00	1.00	0.00	0.87
SGDClassifier	0.00	0.77	0.00	1.00	0.00	0.87
AdaBoostClassifier	0.63	0.79	0.13	0.98	0.21	0.87
RandomForestClassifier	0.64	0.79	0.17	0.97	0.27	0.87

Tabla 6.2: *Evaluación del modelo de dificultad binario ingenuo sobre SQuAD dev entrenado con SQuAD train*

Modelo *SQuAD train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.82	0.43	0.27	0.90	0.41	0.58
GaussianNB	0.64	0.53	0.94	0.12	0.76	0.20
SVC	0.00	0.38	0.00	1.00	0.00	0.55
SGDClassifier	0.00	0.38	0.00	1.00	0.00	0.55
AdaBoostClassifier	0.76	0.39	0.13	0.93	0.22	0.55
RandomForestClassifier	0.78	0.38	0.04	0.98	0.07	0.55

Tabla 6.3: *Evaluación del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con SQuAD train*

En las tablas 6.2 y 6.3 se observa que los modelos *SVC* y *SGDClassifier* no consiguen clasificar ejemplos como difíciles.

Como modelo de predicción de errores sobre *SQuAD*, el planteamiento de regresión logística (*LogisticRegression*) obtiene la exactitud más alta, pero no consigue generalizar sus resultados sobre *NewsQA*. Sin embargo, el modelo predictor binario ingenuo basado en *Naive Bayes* (*GaussianNB*) es el modelo

confianza en caso de acierto, ver su gráfica en 3.11

que más generaliza su resultado para otra colección no empleada para su entrenamiento, como se puede observar en la tabla 6.4.

Modelo *SQuAD train*	Acuruacy *SQuAD train*	Accuracy *SQuAD dev*	Accuracy *NewsQA test*
LogisticRegression	0.89	**0.79**	0.51
GaussianNB	0.86	0.77	**0.62**
SVC	0.88	0.77	0.38
SGDClassifier	0.88	0.77	0.38
AdaBoostClassifier	0.89	0.78	0.43
RandomForestClassifier	1.00	0.78	0.45

Tabla 6.4: *Exactitud (accuracy) del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo sobre SQuAD dev 2.0*

6.3.2 Modelo de predicción de dificultad binario por mayoría

En el planteamiento de dificultad por mayoría no se emplea la confianza de respuesta de ningún Modelo Neuronal de Lenguaje de referencia, el objetivo es predecir en base a la formulación y caracterización la dificultad del ejemplo de la colección de Pregunta-Respuesta. En la sección 5.2 se incluye la distribución de preguntas anotadas por dificultad binaria por mayoría para la obtención de los resultados, ver tablas 6.5 y 6.6.

En este planteamiento, para los resultados entrenados sobre *SQuAD*, el baseline basado en *Naive Bayes* (*GaussianNB*) obtiene la exactitud más baja. Sin embargo, los modelos predictores basados en regresión logística, máquinas de soporte o *RandomForest* mejoran sus resultados para otra colección

Modelo *SQuAD train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.00	0.92	0.00	1.00	0.00	0.96
GaussianNB	0.18	0.92	0.11	0.95	0.14	0.94
SVC	0.00	0.92	0.00	1.00	0.00	0.96
SGDClassifier	0.00	0.92	0.00	1.00	0.00	0.96
AdaBoostClassifier	0.14	0.92	0.00	1.00	0.00	0.96
RandomForestClassifier	0.00	0.92	0.00	1.00	0.00	0.96

Tabla 6.5: *Evaluación del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 para entrenamiento con SQuAD train*

Modelo *SQuAD train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.90	0.57	0.01	1.00	0.01	0.73
GaussianNB	0.44	0.64	0.88	0.16	0.59	0.26
SVC	0.00	0.57	0.00	1.00	0.00	0.72
SGDClassifier	0.00	0.57	0.00	1.00	0.00	0.72
AdaBoostClassifier	0.47	0.57	0.06	0.94	0.11	0.71
RandomForestClassifier	0.00	0.57	0.00	1.00	0.00	0.72

Tabla 6.6: *Evaluación del modelo de dificultad binario sobre NewsQA test entrenado con SQuAD train 2.0*

no empleada para su entrenamiento con respecto a la predicción previa por el planteamiento binario ingenuo de dificultad, como se puede observar en la tabla 6.7.

Las métricas de *accuracy* ascienden en la colección con la misma distribución de entrenamiento (*SQuAD dev*) dado que la precisión de predicción de las preguntas fáciles es alta y es la clase mayoritaria, en este planteamiento de dificultad por mayoría se ha obtenido un número muy reducido de preguntas anotadas como difíciles para la colección *SQuAD dev* (ver tabla 5.4).

Modelo *SQuAD train*	Accuracy *SQuAD train*	Accuracy *SQuAD dev*	Accuracy *NewsQA test*
LogisticRegression	0.97	0.92	**0.57**
GaussianNB	0.94	0.88	0.47
SVC	0.97	0.91	**0.57**
SGDClassifier	0.97	0.92	**0.57**
AdaBoostClassifier	0.97	0.92	0.56
RandomForestClassifier	0.99	0.91	**0.57**

Tabla 6.7: *Exactitud (accuracy) del modelo de dificultad binario por mayoría sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0*

6.3.3 Modelo de predicción de dificultad multiclase

Tras la anotación de dificultad por difícil, fácil y media detallada en la sección 5.3 se ha descrito la distribución de preguntas sobre este criterio. Los resultados de predicción tras aplicar el modelo sobre esta anotación se muestran en las tablas 6.8 y 6.10, y las medidas F1 en las tablas 6.9 y 6.11.

Modelo *SQuAD train*	Precisión Difícil	Precisión Fácil	Precisión Media	Cobertura Difícil	Cobertura Fácil	Cobertura Media
LogisticRegression	0.00	0.68	0.60	0.00	0.98	0.16
GaussianNB	0.12	0.70	0.53	0.05	0.91	0.23
SVC	0.00	0.67	0.66	0.00	0.99	0.11
SGDClassifier	0.00	0.67	0.66	0.00	1.00	0.09
AdaBoostClassifier	0.10	0.69	0.59	0.00	0.97	0.18
RandomForestClassifier	0.00	0.69	0.59	0.00	0.96	0.23

Tabla 6.8: *Evaluación del modelo de dificultad multiclase sobre SQuAD dev 2.0*

Modelo *SQuAD train*	F1-score Difícil	F1-score Fácil	F1-score Media
LogisticRegression	0.00	0.80	0.26
GaussianNB	0.07	0.79	0.32
SVC	0.00	0.80	0.18
SGDClassifier	0.00	0.80	0.16
AdaBoostClassifier	0.00	0.80	0.28
RandomForestClassifier	0.00	0.81	0.33

Tabla 6.9: *F1-score del modelo de dificultad multiclase sobre SQuAD dev*

Modelo *SQuAD train*	Precisión Difícil	Precisión Fácil	Precisión Media	Cobertura Difícil	Cobertura Fácil	Cobertura Media
LogisticRegression	0.00	0.29	0.20	0.00	0.99	0.09
GaussianNB	0.47	0.34	0.33	0.88	0.19	0.00
SVC	0.00	0.28	0.17	0.00	0.99	0.06
SGDClassifier	0.00	0.28	0.15	0.00	1.00	0.04
AdaBoostClassifier	0.55	0.28	0.26	0.18	0.71	0.19
RandomForestClassifier	0.50	0.26	0.50	0.00	1.00	0.00

Tabla 6.10: *Evaluación del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0*

Modelo *SQuAD train*	F1-score Difícil	F1-score Fácil	F1-score Media
LogisticRegression	0.00	0.45	0.12
GaussianNB	0.61	0.25	0.01
SVC	0.00	0.44	0.08
SGDClassifier	0.00	0.43	0.06
AdaBoostClassifier	0.27	0.40	0.22
RandomForestClassifier	0.00	0.41	0.01

Tabla 6.11: *F1-score del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0*

Para los resultados entrenados sobre *SQuAD*, el planteamiento basado en *Naive Bayes* (*GaussianNB*) es el modelo que mejor generaliza sus resultados sobre *NewsQA*. En general, los modelos predictores empeoran sus resultados para esta predicción de dificultad multiclase en comparación con ambos de los planteamientos binarios, como se puede observar en la tabla 6.12.

Modelo *SQuAD train*	Accuracy *SQuAD train*	Accuracy *SQuAD dev*	Accuracy *NewsQA test*
LogisticRegression	0.80	**0.68**	0.28
GaussianNB	0.78	0.65	**0.45**
SVC	0.79	0.67	0.27
SGDClassifier	0.79	0.67	0.27
AdaBoostClassifier	0.80	**0.68**	0.32
RandomForestClassifier	0.99	**0.68**	0.30

Tabla 6.12: *Exactitud (accuracy) del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo de dificultad multiclase sobre SQuAD dev*

6.4 Evaluación de dificultad con modelos con especialización sobre *NewsQA*

Para el entrenamiento de estos modelos predictores de dificultad se ha empleado la colección de *NewsQA* con su partición de entrenamiento sobre las preguntas contestables (57.210) y los Modelos Neuronales de Lenguaje con especialización en *NewsQA* (ver la tabla 5.1). Para sus primeras evaluaciones se ha empleado 3.218 preguntas contestables de *NewsQA test* (ver la tabla 6.13). Para la evaluación de generalización se emplea 5.928 preguntas contestables de *SQuAD dev* (ver la tabla 6.14).

6.4.1 Modelo de predicción binario ingenuo

Para este planteamiento de modelo predictor se considera la variable de confianza arrojada por un Modelo Neuronal de Lenguaje de referencia[2]. En la sección 5.1 se ha descrito la distribución de preguntas sobre este criterio. Los resultados de predicción tras aplicar el modelo sobre esta anotación se muestran en las tablas 6.13 y 6.14.

Modelo *NewsQA train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.97	0.08	0.77	0.43	0.86	0.13
GaussianNB	0.96	0.05	0.58	0.51	0.73	0.09
SVC	0.97	0.08	0.82	0.35	0.88	0.13
SGDClassifier	0.96	0.07	0.86	0.24	0.91	0.11
AdaBoostClassifier	0.97	0.07	0.75	0.44	0.84	0.12
RandomForestClassifier	0.97	0.08	0.77	0.49	0.86	0.14

Tabla 6.13: *Evaluación del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con NewsQA train*

Modelo *NewsQA train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.78	0.62	0.70	0.71	0.74	0.66
GaussianNB	0.84	0.41	0.01	1.00	0.01	0.58
SVC	0.75	0.63	0.74	0.64	0.75	0.64
SGDClassifier	0.75	0.63	0.74	0.65	0.75	0.64
AdaBoostClassifier	0.68	0.53	0.66	0.55	0.67	0.54
RandomForestClassifier	0.74	0.58	0.67	0.65	0.70	0.61

Tabla 6.14: *Evaluación del modelo de dificultad binario ingenuo sobre NewsQA test entrenado con SQuAD dev 2.0*

Contrastando las métricas de exactitud para la predicción de dificultad binaria ingenua sobre *SQuAD*, se observa que para los modelos adaptados en *NewsQA* se obtiene métricas de prediccion de dificultad binaria ingenua

[2]Se ha escogido la confianza de *BERT-base* nuevamente

Modelo *NewsQA train*	Accuracy *NewsQA train*	Accuracy *NewsQA test*	Accuracy *SQuAD dev*
LogisticRegression	0.71	0.76	**0.70**
GaussianNB	0.63	0.58	0.41
SVC	0.71	0.80	**0.70**
SGDClassifier	0.70	**0.83**	**0.70**
AdaBoostClassifier	0.74	0.74	0.62
RandomForestClassifier	0.99	0.75	0.66

Tabla 6.15: *Exactitud (accuracy) del modelo de dificultad binario ingenuo sobre SQuAD test entrenado con NewsQA train. Evaluación del modelo sobre NewsQA test*

en la partición de *SQuAD dev* menores que las obtenidas con los modelos entrenados con *SQuAD* (ver tabla 6.4) a pesar de tener este modelo predictor mayor número de preguntas difíciles para su entrenamiento (mayor número de preguntas difíciles obtenidas en la anotación de *NewsQA train* que *SQuAD train*).

6.4.2 Modelo de predicción de dificultad binario por mayoría

A continuación se puede observar los resultados para los modelos predictores binarios por mayoría entrenados con *NewsQA train* (tablas 6.16 y 6.17), a partir de la distribución de preguntas anotadas por dificultad binaria por mayoría en la sección 5.2.

Se compara la tabla 6.18 con los resultados obtenidos con el planteamiento de predicción binaria ingenua para especialización sobre *NewsQA*:

Modelo *NewsQA train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.68	0.58	0.41	0.81	0.52	0.68
GaussianNB	0.59	0.58	0.54	0.64	0.56	0.61
SVC	0.68	0.57	0.37	0.83	0.48	0.68
SGDClassifier	0.67	0.56	0.32	0.85	0.43	0.67
AdaBoostClassifier	0.73	0.61	0.47	0.83	0.57	0.71
RandomForestClassifier	0.74	0.63	0.51	0.83	0.60	0.72

Tabla 6.16: *Evaluación del modelo de dificultad binario por mayoría sobre NewsQA test entrenado con NewsQA train*

Modelo *NewsQA train*	Precisión Difícil	Precisión Fácil	Cobertura Difícil	Cobertura Fácil	F1-score Difícil	F1-score Fácil
LogisticRegression	0.43	0.82	0.04	0.99	0.07	0.90
GaussianNB	0.67	0.82	0.00	1.00	0.01	0.90
SVC	0.37	0.82	0.02	0.99	0.04	0.90
SGDClassifier	0.35	0.82	0.05	0.98	0.09	0.89
AdaBoostClassifier	0.31	0.83	0.15	0.92	0.21	0.88
RandomForestClassifier	0.35	0.83	0.11	0.96	0.16	0.89

Tabla 6.17: *Evaluación del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 test entrenado con NewsQA train*

se observa mayor exactitud en todos los modelos para la predicción sobre *SQuAD dev*, pero no para la misma distribución de *NewsQA test*.

Considerando los datos de 6.7 para la comparación de la predicción de preguntas con el planteamiento de dificultad por mayoría, se observa en la tabla 6.18 valores superiores de la predicción sobre *NewsQA test* ya que se considera los modelos neuronales adaptados en *NewsQA*.

Modelo *NewsQA train*	Accuracy *NewsQA train*	Accuracy *NewsQA test*	Accuracy *SQuAD dev*
LogisticRegression	0.66	0.61	**0.82**
GaussianNB	0.60	0.59	**0.82**
SVC	0.65	0.60	0.81
SGDClassifier	0.65	0.58	0.81
AdaBoostClassifier	0.71	0.65	0.78
RandomForestClassifier	0.99	**0.66**	0.80

Tabla 6.18: *Exactitud (accuracy) del modelo de dificultad binario por mayoría sobre SQuAD dev 2.0 test entrenado con NewsQA train. Evaluación del modelo de dificultad binario por mayoría sobre NewsQA test*

6.4.3 Modelo de predicción de dificultad multiclase

Tras la anotación de dificultad por difícil, fácil y media y el entrenamiento sobre la colección *NewsQA train* se obtiene los resultados de las tablas 6.19, 6.20, 6.21 y 6.22. En la sección 5.3 se incluye número de preguntas anotadas por dificultad multiclase sobre *NewsQA*.

Modelo *NewsQA train*	Precisión Difícil	Precisión Fácil	Precisión Media	Cobertura Difícil	Cobertura Fácil	Cobertura Media
LogisticRegression	0.65	0.41	0.00	0.67	0.64	0.00
GaussianNB	0.64	0.39	0.20	0.51	0.72	0.06
SVC	0.63	0.42	0.00	0.73	0.57	0.00
SGDClassifier	0.59	0.45	0.00	0.83	0.40	0.00
AdaBoostClassifier	0.70	0.45	0.31	0.65	0.78	0.03
RandomForestClassifier	0.71	0.48	0.38	0.70	0.76	0.06

Tabla 6.19: *Evaluación del modelo de dificultad multiclase sobre NewsQA test entrenado con NewsQA train*

En la tabla 6.23 se observa que los modelos predictores de dificultad entrenados sobre *NewsQA* no generalizan sus resultados para la predicción de

Modelo *NewsQA train*	F1-score Difícil	F1-score Fácil	F1-score Media
LogisticRegression	0.66	0.50	0.00
GaussianNB	0.57	0.51	0.09
SVC	0.67	0.48	0.00
SGDClassifier	0.69	0.43	0.00
AdaBoostClassifier	0.67	0.57	0.05
RandomForestClassifier	0.70	0.59	0.10

Tabla 6.20: *F1-score del modelo de dificultad multiclase sobre NewsQA test*

Modelo *NewsQA train*	Precisión Difícil	Precisión Fácil	Precisión Media	Cobertura Difícil	Cobertura Fácil	Cobertura Media
LogisticRegression	0.37	0.40	0.75	0.18	0.96	0.00
GaussianNB	0.67	0.40	0.53	0.00	0.98	0.11
SVC	0.30	0.40	0.00	0.28	0.84	0.00
SGDClassifier	0.27	0.42	0.00	0.45	0.72	0.00
AdaBoostClassifier	0.32	0.42	0.45	0.20	0.81	0.17
RandomForestClassifier	0.34	0.43	0.46	0.22	0.85	0.14

Tabla 6.21: *Evaluación del modelo de dificultad multiclase sobre SQuAD 2.0 dev entrenado con NewsQA train*

Modelo *NewsQA train*	F1-score Difícil	F1-score Fácil	F1-score Media
LogisticRegression	0.25	0.57	0.00
GaussianNB	0.01	0.57	0.18
SVC	0.29	0.54	0.00
SGDClassifier	0.34	0.53	0.00
AdaBoostClassifier	0.24	0.56	0.25
RandomForestClassifier	0.27	0.57	0.21

Tabla 6.22: *F1-score del modelo de dificultad multiclase sobre SQuAD dev 2.0 entrenado con NewsQA train*

dificultad sobre *SQuAD*. En este planteamiento de anotación por dificultad multiclase se obtienen peores métricas que en el resto de planteamientos de dificultad contemplados para modelos especializados con *NewsQA* en esta sección.

Modelo *NewsQA train*	Accuracy *NewsQA train*	Accuracy *NewsQA test*	Accuracy *SQuAD dev*
LogisticRegression	0.53	0.54	0.40
GaussianNB	0.50	0.48	0.41
SVC	0.52	0.55	0.38
SGDClassifier	0.50	0.55	0.37
AdaBoostClassifier	0.58	0.57	**0.42**
RandomForestClassifier	0.99	**0.59**	**0.42**

Tabla 6.23: *Exactitud (accuracy) del modelo de dificultad multiclase sobre NewsQA test entrenado con SQuAD train 2.0. Evaluación del modelo de dificultad multiclase sobre SQuAD dev 2.0*

Comparando con los datos de la tabla 6.12, se observa que la predicción de preguntas con el planteamiento de dificultad multiclase para *NewsQA test*, los resultados de la tabla 6.23 muestra valores superiores de la predicción ya que se considera los modelos neuronales adaptados en *NewsQA* en esta sección.

Capítulo 7

Conclusiones y trabajos futuros

Resolver la necesidad de acceso a la información de usuarios a través de preguntas arbitrarias sobre gran cantidad de información digital disponible actualmente motiva el estudio, uso y diseño de los sistemas de Pregunta-Respuesta. Buscadores web, repositorios documentales y sistemas de indexación de contenidos hasta los actuales sistemas de diálogo o *chatbots* se benefician de los métodos, arquitecturas, técnicas y mejoras propuestas en el área de Búsqueda de Respuestas a lo largo de su evolución como ámbito de investigación.

Gracias a colecciones como *SQuAD*, *NewsQA* y *RACE* ha sido posible la adaptación de Modelos Neuronales de Lenguaje pre-entrenados, que gracias a su mayor número de parámetros y al uso de corpora masivos de recursos digitales tienen mejores rendimientos que los planteamientos anteriores. Con

ellos se ha superado las limitaciones de representación de secuencias largas de lenguaje mejorando la contextualización y atención entre fragmentos y oraciones. Ejemplos de estos modelos son: *GPT* de OpenAI, *BERT* de Google y sus variaciones *XLNet*, *RoBERTa* y *T5*.

En los estudios es escaso el esfuerzo que se dedica al análisis de errores cometidos por el modelo en función del tipo de pregunta y tipo de respuesta ya que se plantea la inspección de muestras, categorizando fenómenos o características lingüísticas de forma heterogénea, lo que no permite analizar de forma generalizable las colecciones y dificultades que plantean a los modelos en la evaluación con cada colección. Solamente en uno de ellos se observa a partir de modificaciones ruidosas que los Modelos Neuronales de Lenguaje son sensibles a fallos ortográficos y permutaciones a nivel de palabra y oraciones, ante variaciones adversas y presentan errores ante razonamientos lógicos o inferencias.

En este trabajo se proponen varias líneas encaminadas a mitigar este tipo de estudios que realizan comparativas de las colecciones con perspectiva cualitativa por tarea propuesta y origen de sus textos, o con perspectiva cuantitativa por número de preguntas, respuestas y textos asociados.

Se realiza la caracterización de colecciones y el estudio de fallos cometidos por los sistemas observando que en las colecciones de Pregunta-Respuesta extractivo (*SQuAD*, *NewsQA*) predomina las preguntas de hechos concretos con partículas interrogativas *wh* (*what*, *who*, *where*, *when*, etc.) con entidades

reconocidas en sus respuestas (las más frecuentes personas, organizaciones y respuestas numéricas) de forma poco balanceada en cuanto a más tipo de entidades (localizaciones, lugares, porcentajes, cantidades monetarias y más). Con el análisis del foco de la pregunta, se obtiene que no se solicita de forma explícita el tipo de entidad esperada como respuesta. Gracias al análisis sintáctico de las respuestas se observa que también están compuestas por sintagmas nominales, adjetivos y menos comunes sintagmas verbales y preposicionales que suponen mayor dificultad, al igual que las respuestas de mayor longitud de palabras a extraer.

Se plantea una metodología para anotar automáticamente la dificultad de preguntas de una colección en base a la salida de varios sistemas en base a un criterio de comparación de la respuesta aportada por un modelo con la respuesta de referencia (*gold standard*). Esta metodología permite utilizar tres tipos de anotación distintos donde se diferencian las preguntas más fáciles de las más complicadas, que sirve para clasificar colecciones en cuanto a su dificultad. Así, teniendo en cuenta los resultados obtenidos con diversos Modelos Neuronales de Lenguaje se considera una dificultad para cada uno en concreto (clasificación binaria ingenua) o para la mayoría de los modelos (clasificación binaria por mayoría) o multiclase (fácil, media o difícil). Las preguntas de dificiltad media se considera si una respuesta es parcialmente correcta debido a una selección errónea del fragmento de respuesta del texto o si no ha sido fallada por la mitad o mayoría de los modelos contemplados.

Finalmente, se proponen varios modelos de Aprendizaje Automático para predecir la anotación de dificultad propuesta de preguntas sin conocer los resultados previos de otros sistemas. En estos experimentos se ve que es posible predecir la dificultad, aunque todavía hay margen de mejora para realizar esta predicción. Con los resultados obtenidos de exactitud, se observa que predecir dificultad con el planteamiento binario ingenuo es más sencillo que el resto de planteamientos ya que el patrón de dificultad por error para cada modelo con respecto a los errores que cometen los demás es más detectable al emplear la confianza arrojada por el modelo particular.

De este modo, con este trabajo se avanza en los estudios que permiten caracterizar dónde se equivocan los sistemas actuales de Búsqueda de Respuestas y, por tanto, hacia dónde deben encaminar sus esfuerzos para su mejora. Además, se marca una metodología para clasificar las colecciones en cuanto a la dificultad de las preguntas que proponen que permitan su comparativa de forma homogénea y objetiva por particiones de caracterizaciones lingüísticas similares.

Como trabajo futuro se propone incrementar las pruebas sobre variaciones de las colecciones originales formadas con ejemplos adversos mediante generación automática con diversos criterios, para así obtener anotaciones más robustas. Además, se plantea estudiar otros modelos de predicción que utilicen más información para detectar la dificultad de una pregunta, también ante preguntas no contestables.

Bibliografía

C. Aspillaga, A. Carvallo, and V. Araujo. Stress Test Evaluation of Transformer-based Models in Natural Language Understanding Tasks. *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 1882–1894. Marseille, 11–16 May 2020. arXiv:2002.06261*, 2020. URL https://github.com/caspillaga/noisy-squad.

Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. *The journal of Machine Learning Research, 3, 1137-1155*, 2003.

Tom B. Brown, Benjamin Mann, Nick Ryder, and Melanie Subbiah. Language Models are Few-Shot Learners. *Technical report, OpenAI.*, 2020. URL https://arxiv.org/pdf/2005.14165.pdf.

Jose Camacho-Collados and Mohammad Taher Pilehvar. From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research 63 (2018): 743-788*, 2018.

D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*, 2017.

Danqui Chen, Jason Bolton, and Christopher D. Manning. A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. *Computer Science Stanford University*, 2016. URL https://cs.nyu.edu/~kcho/DMQA/.

Peter Clark and Oren Etzioni. My Computer Is an Honor Student — But How Intelligent Is It? Standardized Tests as a Measure of AI. *Association for the Advancement of Artificial Intelligence*, 2016.

Charles L. Clarke, Nick Craswell, and Ian Soboroff. Overview of the trec 2009 web track. *WATERLOO UNIV (ONTARIO)*, 2009.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural Language Processing (almost) from scratch. *Journal of machine learning research, 12(ARTICLE), 2493-2537.*, 2011.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805.*, 2018.

D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, and etc. Gondek, D. Building Watson: An overview of the DeepQA project. *AI magazine, 31(3), 59-79*, 2010.

B. F. Green Jr, A. K. Wolf, C. Chomsky, and K. Laughery. Baseball: an automatic question-answerer. *Western Joint IRE-AIEE-ACM Computer Conference*, 1961.

Matthew Honnibal and Ines Montani. SpaCy. 2016-2021. URL https://spacy.io/.

Daniel Jurafsky and James H. Martin. Speech and Language Processing. An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. *Pearson International Edition. Second and Third Edition.*, 2009-2021. URL https://web.stanford.edu/~jurafsky/slp3/.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale Reading Comprehension Dataset From Examinations. *In Advances in neural information processing systems (pp. 5998-6008).*, 2017. URL http://www.cs.cmu.edu/~glai1/data/race/.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692.*, 2019.

Christopher D. Manning. Lecture 10: (Textual) Question Answering. *Natural Language Processing with Deep Learning. CS224N/Ling284.*, 2020. URL http://web.stanford.edu/class/cs224n/slides/cs224n-2019-lecture10-QA.pdf.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosy. The Stanford CoreNLP Natural Language Processing Toolkit. *In Association for Computational Linguistics (ACL)*, 2014. URL https://stanfordnlp.github.io/CoreNLP/index.html.

Patricio Martínez-Barco, José Luis Vicedo, Estela Saquete, and David Tomás. Sistemas de Pregunta-Respuesta. *Grupo de Procesamiento del Lenguaje y Sistemas de Información, Universidad de Alicante*, 2014.

D. Moldovan, S. Harabagiu, M. Pasca, R. Mihalcea, R.and Girju, R. Goodrum, and V. Rus. The Structure and Performance of an Open-Domain Question Answering System. *In Proceedings of the Conference of the Association for Computational Linguistics (ACL-2000), 563–570.*, 2000.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marciniewicz. Building a large annotated corpus of English: the Penn Treebank. *https://catalog.ldc.upenn.edu/docs/LDC95T7/cl93.html.* URL https://courses.washington.edu/hypertxt/csar-v02/penntable.html.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. URL https://scikit-learn.org/stable/index.html.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Christopher Gardner, Matt Clark, Kenton Lee, and Luke Zettlemoyer†. Deep contextualized word representations. *Allen Institute for Artificial Intelligence and Paul G. Allen School of Computer Science and Engineering, University of Washington*, 2018.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. Universal Dependency Parsing from Scratch. *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, 2018. URL https://stanfordnlp.github.io/stanfordnlp/#citing-stanfordnlp-in-papers.

Peng Qi, Yuhao Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. *Association for Computational Linguistics (ACL) System Demonstrations*, 2020. URL https://stanfordnlp.github.io/stanza/index.html.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding with unsupervised learning. *Technical report, OpenAI.*, 2018. URL https://openai.com/blog/language-unsupervised/.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. *Technical report, OpenAI.*, 2019. URL https://openai.com/blog/better-language-models/.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, and P. J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv preprint arXiv:1910.10683.*, 2019.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *Computer Science Department. Stanford University.*, 2016. URL https://rajpurkar.github.io/SQuAD-explorer/explore/1.1/dev/.

Pranav Rajpurkar, Roin Jia, and Percy Liang. Know What You Don't Know: Unanswarable Questions for SQuAD. *Computer Science Department. Stanford University.*, 2018. URL https://rajpurkar.github.io/SQuAD-explorer/.

Siva Reddy, Danqui Chen, and Christopher D. Manning. CoQA: A Conversational Question Answering Challenge. *Computer Science Department. Stanford University.*, 2019. URL https://stanfordnlp.github.io/coqa/.

M. Richardson, C. J. Burges, and E Renshaw. MCTest: A challenge dataset for the open-domain machine comprehension of text. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (pp. 193-203)*, 2013. URL https://mattr1.github.io/mctest/data.html.

A. Rodrigo and A. Peñas. A study about the future evaluation of Question-Answering systems. *Knowledge-Based Systems, 137, 83-93*, 2017.

Álvaro Rodrigo, Jesús Herrera, and Anselmo Peñas. The Effect of Answer Validation on the Performance of Question-Answering Systems. *Elsevier*, 2018.

Anna Rogers, Olga Kovaleva, Matthew Downey, and Anna Rumshisky. Getting Closer to AI Complete Question Answering: A Set of Prerequisite Real Tasks. *University of Massachusetts Lowell. Proceedings of the AAAI Conference on Artificial Intelligence.*, 2020. URL http://text-machine.cs.uml.edu/projects/quail/.

Sebastian Ruder. Transfer Learning - Machine Learning's Next Frontier, 2017. URL http://ruder.io/transfer-learning/.

Sebastian Ruder. NLP's ImageNet moment has arrived. *The Gradient*, 2018a. URL https://thegradient.pub/nlp-imagenet/.

Sebastian Ruder. A Review of the Neural History of Natural Language Processing, 2018b. URL http://ruder.io/a-review-of-the-recent-history-of-nlp/.

Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. Transfer Learning in Natural Language Processing. In *Proceedings of the 2019*

Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials, pages 15–18, 2019. URL https://ruder.io/state-of-transfer-learning-in-nlp/.

Sunita Sarawagi. Information Extraction. *Foundations and Trends in Databases 1.3, 261-377*, 2008.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. *En New methods in language processing. TreeTagger - a part-of-speech tagger for many languages*, 2013. URL http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/.

M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

A. Talmor and J. Berant. MultiQA: An empirical investigation of generalization and transfer in Reading Comprehension. *arXiv preprint arXiv:1905.13453*, 2019.

Adam Trischler, Tong Wang, Xingdi Yuan, et al. NewsQA: A machine comprehension Dataset. *arXiv preprint arXiv:1611.09830v3*, 2017.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, and Gomez. Attention is all you need. *In Advances in neural information processing systems (pp. 5998-6008).*, 2017.

José Luis Vicedo. La Búsqueda de Respuestas: Estado Actual y Perspectivas. *Departamento de Lenguajes y Sistemas Informáticos, Universidad de Alicante*, 2004.

Alex Wang, Amanpreet Singh, Julian Michael, and Felix Hill. GLUE: A multi-task benchmark and analysis platform for Natural Language Understanding. *Courant Institute of Mathematical Sciences, New York University and Paul G. Allen School of Computer Science and Engineering, University of Washington and Deep Mind*, 2018. URL https://gluebenchmark.com/,https://www.quora.com/q/quoradata/First-Quora-Dataset-Release-Question-Pairs.

Xiaozhi Wang and Zhengyan Zhang. PLMpapers (Bertology). 2019-2021. URL https://github.com/thunlp/PLMpapers.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, and Pierric Cistac. HuggingFace's Transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771. https://huggingface.co/transformers/*, 2019. URL https://huggingface.co/models.

William A. Woods. Progress in Natural Language Understanding: an application to lunar geology. *Proceedings of the June 4-8, 1973, National Computer Conference and Exposition*, 1973.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le. XLNet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237.*, 2019.

Tom Youngy, Devamanyu Hazarikaz, Soujanya Poria, and Erik Cambria. Recent Trends in Deep Learning Based Natural Language Processing. *ieee Computational intelligenCe magazine 13.3 (2018): 55-75*, 2018.

Apéndice A

Anexo 1. Fases y desarrollo del trabajo

A continuación se detalla las fases seguidas para la realización de los objetivos y obtención de resultados y conclusiones expuestas en este trabajo. También, los principales desarrollos y funciones en código Python ejecutado en los entornos *Jupyter Notebook* y *Google Colab* con GPU, disponible en Github.

A.1 Fases del trabajo

1. Estudio del estado del arte sobre Búsqueda de Respuestas, sus colecciones y Modelos Neuronales de Lenguaje.

2. Estudio de los fenómenos lingüísticos y dificultades en el estado del estudio del arte.

3. Obtención de las colecciones de Pregunta-Respuesta de *SQuAD*, *NewsQA* (y noticias *CNN*) y *RACE*. Para ello, se ha realizado su carga desde los formatos de origen (JSON, CSV, etc.) y relación de preguntas con sus textos (caso *NewsQA* y *CNN Daily News*) y recopilación de particiones de entrenamiento, test, desarrollo o por dificultad (*RACE*).

4. Desarrollo para tareas de Procesamiento de Lenguaje Natural (análisis sintáctico, reconocimiento de entidades y obtención de foco de la pregunta) para su lanzamiento sobre las colecciónes de Pregunta-Respuesta seleccionadas.

5. Análisis de patrones obtenidos con la caracterización lingüística de las colecciones de Pregunta-Respuesta (dificultad *a priori*).

6. Lanzamiento de 9 Modelos Neuronales de Lenguaje pre-entrenados disponibles públicamente seleccionados por su *fine-tunning* y basados en *BERT*, *RoBERTa* y *T5* sobre las colecciones de pregunta-respuesta extractivo *SQuAD* y *NewsQA*.

7. Estudio de dificultad en base a los errores cometidos por los Modelos Neuronales de Lenguaje pre-entrenados (dificultad *a posteriori*).

8. Anotación para las colecciones *SQuAD* y *NewsQA* por dificultad para cada planteamiento: clasificación binaria ingenua (fácil, difícil al menos para un modelo), binaria por mayoría (fácil, difícil) y multiclase (fácil, media, difícil).

9. Diseño y generación de variables para de la pregunta, respuesta, con-

texto y modelo en base a los fenómenos lingüísticos contemplados y errores cometidos.

10. Planteamiento y entrenamiento de un modelo predictor de dificultad a partir de una pregunta, contexto y respuesta. Importancia de las variables para el modelo predictivo con perspectiva de análisis de datos.

11. Evaluación de los modelos de predicción de dificultad sobre la colección de entrenamiento y sobre otra colección no empleada para el entrenamiento para comparar resultados y capacidad de generalización del modelo predictor de dificultad.

A.2 Desarrollo

Se ha empleado diversas librerías abiertas (*open source*) para la carga, procesado de estructuras y datos (JSON y *dataframe* de *pandas*), para Procesamiento de Lenguaje Natural (para análisis sintáctico, reconocimiento de entidades y obtención de foco de la pregunta), para Aprendizaje Automático supervisado (*scikit-learn*) y para Deep Learning en Pregunta-Respuesta, en base a los modelos públicos de *HuggingFace*.

A.2.1 Análisis sintáctico

El etiquetador *TreeTagger* del trabajo Schmid (2013) permite obtener de etiquetas léxicas y el lema de cada palabra de la frase recibida en su entrada. Desarrollado

por Helmut Schmid, Universidad de Stuttgart, este etiquetador permite con árboles de decisión (ID3) adaptar el contexto (no solo bigramas y trigramas) y aprender reglas a partir de la estructura del árbol obtenido, basándose en las etiquetas de *Penn Treebank* (P. Marcus et al.). Se ha empleado su versión pre-entrenada pública para el inglés, con el siguiente desarrollo.

```
1       import treetaggerwrapper
2       tagger = treetaggerwrapper.TreeTagger(TAGLANG='en')
3       def pos_tagging(text, max_length=1000, tagger=tagger):
4             tags = tagger.tag_text(text)
5             results = treetaggerwrapper.make_tags(tags)
6             return results
7
8         # Ejemplos
9       print(pos_tagging('When was the Tower Theatre built?'))
10      >> [Tag(word='When', pos='WRB', lemma='when'), Tag(word='was', pos='VBD',
            lemma='be'), Tag(word='the', pos='DT', lemma='the'), Tag(word='Tower',
            pos='NP', lemma='Tower'), Tag(word='Theatre', pos='NP', lemma='Theatre'
            ), Tag(word='built', pos='VVD', lemma='build'), Tag(word='?', pos='SENT
            ', lemma='?')]
11          print(pos_tagging('1939'))
12      >> [Tag(word='1939', pos='LS', lemma='@card@')]
13      print(pos_tagging('Which name is also used to describe the Amazon
            rainforest in English?'))
14      >> [Tag(word='Which', pos='WDT', lemma='which'), Tag(word='name', pos='NN',
             lemma='name'), Tag(word='is', pos='VBZ', lemma='be'), Tag(word='also',
             pos='RB', lemma='also'), Tag(word='used', pos='VVN', lemma='use'), Tag
            (word='to', pos='TO', lemma='to'), Tag(word='describe', pos='VV', lemma
            ='describe'), Tag(word='the', pos='DT', lemma='the'), Tag(word='Amazon'
            , pos='NP', lemma='Amazon'), Tag(word='rainforest', pos='NN', lemma='
            rainforest'), Tag(word='in', pos='IN', lemma='in'), Tag(word='English',
             pos='NP', lemma='English'), Tag(word='?', pos='SENT', lemma='?')]
15      print(pos_tagging_text('also known in English as Amazonia or the Amazon
            Jungle,'))
16      >> [Tag(word='also', pos='RB', lemma='also'), Tag(word='known', pos='VVN',
            lemma='know'), Tag(word='in', pos='IN', lemma='in'), Tag(word='English'
            , pos='NP', lemma='English'), Tag(word='as', pos='IN', lemma='as'), Tag
            (word='Amazonia', pos='NP', lemma='Amazonia'), Tag(word='or', pos='CC',
             lemma='or'), Tag(word='the', pos='DT', lemma='the'), Tag(word='Amazon'
            , pos='NP', lemma='Amazon'), Tag(word='Jungle', pos='NN', lemma='jungle
            '), Tag(word=',', pos=',', lemma=',')]
17      print(pos_tagging('Jay Z and Beyonce attended which event together in
            August of 2011?'))
18      >> [Tag(word='Jay', pos='NP', lemma='Jay'), Tag(word='Z', pos='NP', lemma='
```

```
        Z'), Tag(word='and', pos='CC', lemma='and'), Tag(word='Beyonce', pos='
        NP', lemma='Beyonce'), Tag(word='attended', pos='VVN', lemma='attend'),
         Tag(word='which', pos='WDT', lemma='which'), Tag(word='event', pos='NN
        ', lemma='event'), Tag(word='together', pos='RB', lemma='together'),
        Tag(word='in', pos='IN', lemma='in'), Tag(word='August', pos='NP',
        lemma='August'), Tag(word='of', pos='IN', lemma='of'), Tag(word='2011',
         pos='CD', lemma='@card@'), Tag(word='?', pos='SENT', lemma='?')]
print(nlp.ner(preprocess_text('MTV Video Music Awards')))
>> [Tag(word='MTV', pos='NP', lemma='MTV'), Tag(word='Video', pos='NP',
    lemma='Video'), Tag(word='Music', pos='NP', lemma='Music'), Tag(word='
    Awards', pos='VVZ', lemma='award')]
```

Listing A.1: *POSTagging con TreeTagger. Ejemplos de POSTagging con TreeTagger 5725dd7d89a1e219009abfeb 56bea8463aeaaa14008c91a9 y 5725b81b271a42140099d097 de SQuAD*

A.2.2 Reconocimiento de entidades

Para el reconocimiento de entidades se ha empleado el servidor de *Stanford CoreNLP* (Manning et al., 2014) encapsulado en Python.

```
from stanfordcorenlp import StanfordCoreNLP
nlp = StanfordCoreNLP('http://localhost', port=9000)

# Preprocesado del texto
import re
def preprocess_text(text_str):
      regular_expr = re.compile('\n|\r|\t|\(|\)|\[|\]|:|\,|\;|"|\?|\-|\%')
      text_str = re.sub(regular_expr, ' ', text_str)
      token_list = text_str.split(' ')
      token_list = [element for element in token_list if element]
      return ' '.join(token_list)

# Filtrado y composicion multipalabra para NER
def filter_ner_relevant(tuple_list):
      ner_dictionary={}
      previous_ner='O'
      for element in tuple_list:
          if element[1] != 'O':
              if element[1] == previous_ner:
                  ner_dictionary[element[1]][-1] += ' ' + element[0]
```

```
                elif element[1] in ner_dictionary.keys():
                    ner_dictionary[element[1]].append(element[0])
                else:
                    ner_dictionary[element[1]] = [element[0]]
            previous_ner = element[1]
        return ner_dictionary

# Ejemplos
print(nlp.ner(preprocess('When was the Tower Theatre built?')))
>> [('When', 'O'), ('was', 'O'), ('the', 'O'), ('Tower', 'O'), ('Theatre', 'O'), ('built', 'O')]
print(filter_ner_relevant(nlp.ner(preprocess_text('1939'))))
>> {'DATE': ['1939']}
print(filter_ner_relevant(nlp.ner(preprocess_text(context_tower_example))))
>> {'DATE': ['1939', 'today', '1916', 'now'],
'PERSON': ['Wishon'], # ERROR
'CITY': ['Tower', 'Tower', 'Fresno', 'Fresno', 'Tower'], 'LOCATION': ['District', 'District', 'District'],
'NUMBER': ['one', 'one'],
'ORGANIZATION': ['Fresno City College','Fresno Normal School','California State University','Fresno City College'],
'MISC': ['World', 'II'], 'CAUSE_OF_DEATH': ['War']}

print(nlp.ner(preprocess('Which name is also used to describe the Amazon rainforest in English?')))
>> {'LOCATION': ['Amazon'], 'NATIONALITY': ['English']}
print(filter_ner_relevant(nlp.ner(preprocess_text('also known in English as Amazonia or the Amazon Jungle,'))))
>> {'NATIONALITY': ['English'], 'LOCATION': ['Amazonia', 'Amazon Jungle']}
print(filter_ner_relevant(nlp.ner(preprocess_text(amazon_context_example))))
>> {'ORGANIZATION': ['Amazon', 'Amazon Jungle', 'Amazon'], # ERROR
'NATIONALITY': ['Portuguese', 'Spanish', 'French', 'Dutch', 'English', 'French'],
'PERSON': ['Selva Amazonica Amazonia', 'Amazoneregenwoud'], # ERROR
'LOCATION': ['Amazonia', 'Amazonia', 'Amazon', 'South'],
'COUNTRY': ['America', 'Brazil', 'Peru', 'Venezuela Ecuador Bolivia Guyana Suriname'],
'NUMBER': ['7 000 000', '2 700 000', '5 500 000', '2 100 000', 'nine', '60', '13', '10', 'four', '390 billion', '16 000']}

print(nlp.ner(preprocess('Jay Z and Beyonce attended which event together in August of 2011?')))
>> {'PERSON': ['Jay Z', 'Beyonce'], 'DATE': ['August of 2011']}
print(filter_ner_relevant(nlp.ner(preprocess_text('MTV Video Music Awards'))))
>> {'ORGANIZATION': ['MTV']}
print(filter_ner_relevant(nlp.ner(preprocess_text(beyonce_context))))
>> {'DATE': ['August', '2011', 'Tonight', 'the week of August 29 2011'],
'PERSON': ['Beyonce', 'Beyonce'], # OK con tilde y sin tilde
'TIME': ['evening'], 'DURATION': ['year'],
```

```
61      'ORGANIZATION': ['MTV', 'Twitter'],
62      'NUMBER': ['12.4 million', '8 868'],
63      'MISC': ['Guinness World Records', 'Googled'],
64      'ORDINAL': ['second', 'second'] # ERROR
65      }
```

Listing A.2: *NER con Stanford CoreNLP*

A.2.3 Obtención de foco de la pregunta

Para la obtención del foco de la pregunta, se emplea resultados obtenidos con los desarrollos anteriores.

```
1       def obtener_foco(query, query_pos):
2           candidate_focus = []
3           minor_index = []
4           if len(query) > 0:
5               if (query[0].lower() == 'who') and 'WP' in query_pos:
6                   candidate_focus.append('person')
7               if 'WP$' in query_pos[0]:
8                   candidate_focus.append('person')
9               if (query[0].lower() == 'where') and 'WRB' in query_pos:
10                  candidate_focus.append('place')
11              if (query[0].lower() == 'when') and 'WRB' in query_pos:
12                  candidate_focus.append('time')
13
14              if sublist(['NN'],query_pos):
15                  minor_index.append(query_pos.index('NN'))
16              if sublist(['NNS'],query_pos):
17                  minor_index.append(query_pos.index('NNS'))
18              if sublist(['NPS'],query_pos):
19                  minor_index.append(query_pos.index('NPS'))
20              if sublist(['NP'],query_pos):
21                  minor_index.append(query_pos.index('NP'))
22
23              if sublist(['WP'], query_pos) and not sublist(['NN'], query_pos)
                    and not sublist(['NNS'], query_pos) and not sublist(['NP'],
                    query_pos) and not sublist(['NPS'], query_pos):
24                  if sublist(['VVG'], query_pos):
25                      minor_index.append(query_pos.index('VVG'))
26                  if sublist(['JJ'], query_pos):
27                      minor_index.append(query_pos.index('JJ'))
28                  if sublist(['VVN'], query_pos):
```

```
                    minor_index.append(query_pos.index('VVN'))
                if sublist(['VVD'], query_pos):
                    minor_index.append(query_pos.index('VVD'))
                if sublist(['RB'], query_pos):
                    minor_index.append(query_pos.index('RB'))

            if len(minor_index) > 0 and min(minor_index) < len(query) and min(
                minor_index) >= 0:
                candidate_focus.append(query[min(minor_index)])

            if ('how much' in ' '.join(query).lower()) or ('how many' in ' '.
                join(query).lower()):
                candidate_focus.append('quantity')

        if candidate_focus:
            return candidate_focus[0]
        else:
            return ''
```

Listing A.3: *Obtención del foco de la pregunta a partir de análisis sintáctico*

A.2.4 Desarrollo para ejecución de modelos sobre colecciones de Pregunta-Respuesta

Se ha empleado la librería abierta de HuggingFace (Wolf et al., 2019) ya que ofrece una interfaz que abstrae la carga y ejecución de modelos de Pregunta-Respuesta, configuración y preprocesado de cada modelo neuronal de lenguaje pre-entrenados como BERT, RoBERTa y T5 para pregunta-respuesta extractivo.

```
    from transformers import pipeline, AutoTokenizer,
        AutoModelForQuestionAnswering

    MODEL = 'phiyodr/bert-base-finetuned-squad-v2'

    tokenizer = AutoTokenizer.from_pretrained(MODEL)
    model = AutoModelForQuestionAnswering.from_pretrained(MODEL) #.to_device('
        cuda')
    qa_specific = pipeline('question-answering', model=model, tokenizer=
        tokenizer) #, device=0

```

```
9       # Ejemplos
10      print(qa_specific(question='When was the Tower Theatre built?', context=
            context_example)
11      >> {'score': 0.9739360213279724, 'start': 184, 'end': 188, 'answer': '1939'
            }
12
13      print(qa_specific(question='Which name is also used to describe the Amazon
            rainforest in English?', context=amazon_context_example)
14      >> {'score': 0.6646409034729004, 'start': 201, 'end': 230, 'answer': '
            Amazonia or the Amazon Jungle'}
15
16      print(qa_specific(question='Jay Z and Beyonce attended which event together
             in August of 2011?', context=beyonce_context)
17      >> {'score': 0.5714967846870422, 'start': 40, 'end': 62, 'answer': 'MTV
            Video Music Awards'}
```

Listing A.4: *Ejemplo de carga y ejecución de un modelo pre-entrenado de HuggingFace para Pregunta-Respuesta*

A.2.5 Desarrollo para asociación foco de la pregunta y entidad de la respuesta

Se propone una comprobación automática de primer nivel mediante la verificación de relaciones entre el foco de la pregunta y respuesta válida en base a reglas de asociación generales. Para ello, se emplean palabras literales del foco y su asociación con las etiquetas de entidades reconocidas enriquecidas ante plurales y algunos sinónimos encontrados en las colecciones.

```
1       def validate_foco_ner(foco, ner_query, answer):
2           result = 'KO'
3           foco_pos = get_pos(str(pos_tagging(foco)))
4           if foco_pos:
5               foco_pos = foco_pos[0]
6           if not isinstance(ner_query, list):
7               ner_query = str(ner_query).replace('[','').replace(']','').replace(
                    "'", '').split(', ')
8           if ner_query == '[]':
9               result = 'NA'
```

```
10          elif ner_query == []:
11              result = 'NA'
12          elif not foco or foco == 'NaN':
13              result = 'NA'
14          elif str(answer)!='' and str(answer)!='NaN' and str(answer)!='[NaN]':
15              if (foco.lower() in ['person','name','people', 'names','
                    nationalities', 'nationality'] or foco_pos in ['NP','NPS']) and
                     sublist(ner_query,['PERSON', 'ORGANIZATION', 'TITLE', '
                    NATIONALITY']):
16                  result = 'OK-PERSON-ORG'
17              if (foco.lower() in ['place', 'country', 'city', 'state', 'province
                    ', 'location', 'area', 'region',
18                          'areas','locations','states', 'cities', 'countries']
                                or foco_pos in ['NP','NPS']) and sublist(ner_query
                                ,['CITY', 'COUNTRY', 'LOCATION', '
                                STATE_OR_PROVINCE']):
19                  result = 'OK-LOC'
20              if (foco.lower() in ['time', 'duration', 'age', 'year', 'month', '
                    day', 'week', 'hour', 'decade', 'century',
21                          'days', 'years', 'hours', 'ages', 'weeks', 'decades',
                                'months', 'centuries']) and sublist(ner_query,['
                                DATE', 'TIME', 'DURATION', 'NUMBER']):
22                  result = 'OK-TIME'
23              if (foco.lower() in ['titles','title','role','roles']) and sublist(
                    ner_query,['TITLE']):
24                  result = 'OK-TITLE'
25              if (foco.lower() in ['percentage']) and sublist(ner_query,['PERCENT
                    ']):
26                  result = 'OK-PERCENT'
27              if (foco.lower() in ['number','numbers','quantity', 'money', 'age',
                     'percentage'] or foco_pos in ['CD','LS', 'NNS']) and sublist(
                    ner_query,['NUMBER', 'PERCENT', 'MONEY', 'ORDINAL', 'CARDINAL'
                    ]):
28                  result = 'OK-NUMBER'
29
30              if foco and sublist([foco.upper()], ner_query):
31                  result = 'OK-' + foco.upper()
32              elif foco and foco[-1]=='s' and len(foco) > 2 and sublist([foco
                    [:-1].upper()], ner_query):
33                  result = 'OK-' + foco[:-1].upper()
34          else:
35              result='NA'
36          return result
```

Listing A.5: *Código de reglas de asociación entre foco de la pregunta y tipo de respuesta con análisis sintáctico y reconocimiento de entidades*

A.2.6 Desarrollo para diferencia entre respuestas y detección de errores

Se propone el criterio de comparación entre la respuesta de referencia y la aportada por un modelo de Búsqueda de Respuestas con el retirado de partículas sin significado *the* y normalización de símbolos de puntuación en ambos textos de respuesta previamente a su comparación. También se valora la respuesta como correcta si la respuesta aportada está contenida en ella, es decir, las palabras coinciden literalmente de forma consecutiva (*word matching*).

```
1      def correct(answer, model_answer, plausible):
2          answer = str(answer).replace("'", '').replace('"', '').replace(',','')
3          model_answer = str(model_answer).replace("'", '').replace('"', '').
               replace(',','')
4          plausible = str(plausible).replace("'", '').replace('"', '').replace(',
               ','').replace('.','')
5          if answer and model_answer:
6              if answer == model_answer:
7                  return True
8              if str(answer).lower().replace('the ', '') == str(model_answer).
                   lower().replace('the ', ''):
9                  return True
10             if str(answer).lower() in str(model_answer).lower() or str(
                   model_answer).lower() in str(answer).lower():
11                 return True
12         elif plausible and model_answer:
13             if plausible == model_answer:
14                 return True
15             if str(plausible).lower().replace('the ', '') == model_answer.lower
                   ().replace('the ', ''):
16                 return True
17             if str(plausible).lower() in str(model_answer).lower() or str(
                   model_answer).lower() in str(plausible).lower():
18                 return True
19         return False
```

Listing A.6: *Desarrollo para comparación de respuesta gold standard y respuesta dada por un modelo*

Se considera parcialmente correcta una pregunta (para caracterizar dificultad media) aquella que ha sido respondida parcialmente, es decir, parte de las palabras en la respuesta son aceptadas por una persona ya que la respuesta aportada está contenida en la correcta o viceversa, es decir, hay selección errónea al comienzo o en la finalización del fragmento extraído como respuesta por un modelo de Búsqueda de Respuestas.

```
1     def correct_medium(answer, model_answer, plausible):
2         answer = answer.replace("'", '').replace('"', '').replace(',','')
3         model_answer = model_answer.replace("'", '').replace('"', '').replace('
              ,','')
4         plausible = plausible.replace("'", '').replace('"', '').replace(',','')
5
6         if answer and model_answer:
7             if answer == model_answer:
8                 return 'FACIL'
9             if str(answer).lower().replace('the ', '') == str(model_answer).
                  lower().replace('the ', ''):
10                return 'FACIL'
11            if str(answer).lower() in str(model_answer).lower() or str(
                  model_answer).lower() in str(answer).lower():
12                return 'MEDIA'
13        elif plausible and model_answer:
14            if plausible == model_answer:
15                return 'FACIL'
16            if str(plausible).lower().replace('the ', '') == model_answer.lower
                  ().replace('the ', ''):
17                return 'FACIL'
18            if str(plausible).lower() in str(model_answer).lower() or str(
                  model_answer).lower() in str(plausible).lower():
19                return 'MEDIA'
20        return 'DIFICIL'
```

Listing A.7: *Desarrollo para comparación de respuesta gold standard y respuesta dada por un modelo, incluyendo dificultad media*

A.2.7 Desarrollo para modelos de predicción

Se ha empleado modelos de Aprendizaje Automático disponibles en la librería *Scikit-learn* (Pedregosa et al., 2011) realizando una comparación entre diversas estrategias para clasificación (batalla de modelos): regresión logística (*LogisticRegression*), probabilísticas con *Naive Bayes* (*GaussianNB*), máquinas de vectores de soporte (*SVC*), descenso de gradiente (*SGDClassifier*) y aprendizaje en conjunto por *bagging* con árboles de decisión (*Random Forest*), y el meta clasificador *AdaBoost* (*AdaBoostClassifier*).

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from sklearn.utils import shuffle
from sklearn.metrics import classification_report, confusion_matrix

# Ejemplo de modelo
from sklearn.linear_model import LogisticRegression

# (...) Carga de datos (matriz X) y variable objetivo (y) de distintas
    particiones
# x_train, x_dev, x_test, y_train, y_dev, y_test

# Normalizacion minimo maximo de ejemplo, se aplica sobre todas las
    particiones
# Carga del scaler preajustado (MinMaxScaler)
x_train = scaler.transform(x_train)
x_train, y_train = shuffle(x_train, y_train, random_state=0)

# Entrenamiento
logisticRegr = LogisticRegression(solver='liblinear')
logisticRegr.fit(x_train, y_train)

# Ejemplo de evaluacion
predictions = logisticRegr.predict(x_test)
score = logisticRegr.score(x_test, y_test)
print(score)
cm = confussion_matrix(y_test, predictions)
print(classification_report(y_test, predictions))
```

Listing A.8: *Entrenamiento y validación con modelo de Scikit-learn*

Apéndice B

Anexo 2. Detalle de caracterización de colecciones de Pregunta-Respuesta

A continuación se tiene detalle sobre el análisis sintáctico, reconocimiento de entidades, ejemplos de fenómenos lingüísticos sobre las colecciones de Búsqueda de Respuestas analizadas: *SQuAD*, *NewsQA* y *RACE*. También más detalle de los errores cometidos por los Modelos Neuronales de Lenguaje contemplados: *BERT*, *RoBERTa* y *T5* en sus versiones de *HuggingFace*.

B.1 Reconocimiento de entidades sobre colecciones de Pregunta-Respuesta

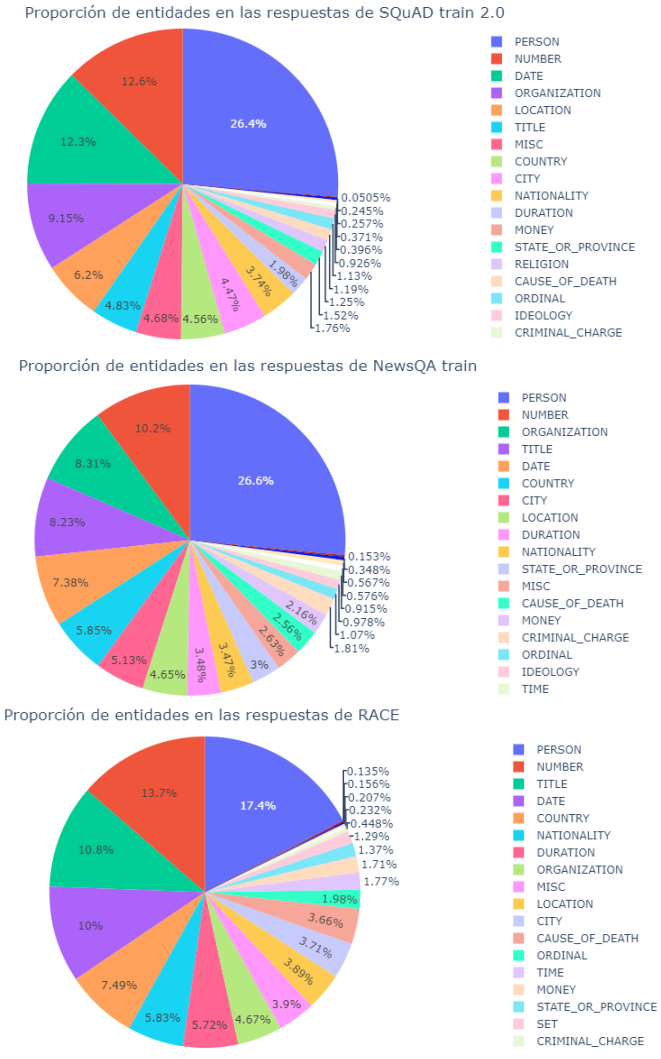


Figura B.1: *Tipo de entidades reconocidas en las respuestas de las colecciones SQuAD, NewsQA y RACE*

Tipo de Entidad	Preguntas *SQuAD train*	Preguntas *SQuAD dev*	Preguntas *NewsQA train*	Preguntas *NewsQA test+dev*	Preguntas *RACE train*	Preguntas *RACE test+dev*
PERSON	16.057	1.150	11.229	1.546	10.707	969
ORGANIZATION	**10.351**	890	3.465	466	1.748	124
CITY	5.468	550	1.346	181	1.099	80
COUNTRY	**7.645**	583	2.813	444	1.311	106
LOCATION	**6.612**	655	1.377	202	1.195	105
NATIONALITY	**5.809**	519	1.516	241	1.031	97
DATE	**11.753**	919	2.967	398	2.755	97
DURATION	**5.536**	437	1.548	222	1.298	97
TIME	233	5	194	21	425	42
PERCENT	36	6	104	10	12	1
NUMBER	5.416	591	2.505	315	1.852	157
ORDINAL	3.163	232	760	100	801	60
MONEY	180	18	295	38	137	153
MISC	7.592	514	1.427	190	1.193	114
URL	15	0	27	2	44	7
CAUSE OF DEATH	2.251	261	1.583	232	600	44
STATE OR PROVINCE	1.697	125	848	110	377	29
CRIMINAL CHARGE	484	22	561	93	93	6
RELIGION	1.831	165	215	27	48	1
TITLE	6.880	565	4.238	649	3.933	356
SET	543	38	123	23	288	38
IDEOLOGY	931	81	275	47	25	3

Tabla B.1: *Número de entidades reconocidas con CoreNLP en las preguntas de las colecciones de Pregunta-Respuesta SQuAD 2.0, NewsQA y RACE*

Tipo de Entidad	Respuestas *SQuAD train*	Respuestas *SQuAD dev*	Respuestas *NewsQA train*	Respuestas *NewsQA test+dev*	Respuestas *RACE train*	Respuestas *RACE test+dev*
PERSON	10.445	686	11.433	1.717	20.973	1.671
ORGANIZATION	3.626	258	3.577	490	5.617	370
CITY	1.771	144	2.209	319	4.461	281
COUNTRY	1.807	133	2.517	406	**9.014**	661
LOCATION	2.458	193	2.003	300	4.686	291
NATIONALITY	1.482	127	1.493	248	**7.017**	559
DATE	4.875	471	3.178	489	**12.034**	906
DURATION	783	51	1.500	213	**6.885**	504
TIME	97	3	394	44	**2.136**	145
PERCENT	102	8	248	44	188	6
NUMBER	5.007	497	4.373	772	**16.443**	1.334
ORDINAL	448	37	461	55	2.382	164
MONEY	698	44	931	145	**2.059**	153
MISC	1.855	130	1.134	150	4.694	317
URL	20	0	66	3	279	20
CAUSE OF DEATH	471	62	1.100	186	4.411	292
STATE OR PROVINCE	601	55	1.291	184	1.647	94
CRIMINAL CHARGE	157	10	780	123	539	52
RELIGION	495	42	244	47	249	11
TITLE	1.914	135	3.544	542	**12.963**	1.054
SET	147	20	150	28	1.559	132
IDEOLOGY	367	33	421	54	163	13

Tabla B.2: *Número por tipo de entidades reconocidas con CoreNLP en las respuestas de las colecciones de Pregunta-Respuesta SQuAD 2.0, NewsQA y RACE*

B.2 Ejemplos de fenómenos lingüísticos

A continuación se detallan ejemplos de preguntas, respuestas y fragmentos anotados como difíciles automáticamente por las diferencias entre la respuesta dada por una persona y los Modelos Neuronales de Lenguaje. Se han seleccionado de la colección *SQuAD* por su caracterización y fenómenos lingüísticos presente en ellos.

B.2.1 Ejemplo de fenómeno de contextualización errónea por escasez de términos comunes

West has additionally appeared and ***participated in*** *many fundraisers, benefit concerts, and has done community work for Hurricane Katrina relief, the* ***Kanye West*** *Foundation, the Millions More Movement, 100 Black Men of America, a Live Earth concert benefit, World Water Day rally and march, Nike runs, and a MTV special helping young Iraq War veterans who struggle through debt and PTSD a second chance after returning home.*

Pregunta: *What are some* ***charitable efforts Kanye west*** *has* ***participated in****?*

Respuesta: *100 Black Men of America, a Live Earth concert benefit, World Water Day rally*

Respuesta *phiyodr/bert-base-finetuned-squad2*: -.

Respuesta *phiyodr/bert-large-finetuned-squad2*: -.

Respuesta *deepset/roberta-base-squad2*: -.

Respuesta *phiyodr/roberta-large-finetuned-squad2*: -.

Respuesta *valhalla/t5-base-squad*: *the Kanye West Foundation, the Millions More Movement, 100 Black Men of America,*.

B.2.2 Ejemplo de confusión por términos comunes y/o coreferencia

In August 2008, West revealed plans to ***open 10*** *Fatburger restaurants in the Chicago area; the first was set to open in September 2008 in Orland Park. The second followed in January 2009, while a third location is yet to be revealed, although the process is being finalized. His company, KW Foods LLC, bought the rights to the chain in Chicago. Ultimately, in 2009, only* ***two locations actually opened****. In February 2011, West shut down the Fatburger located in Orland Park. Later that year, the remaining Beverly location also was shuttered..*

Pregunta: *How many restaurants did Kanye* ***open****?*

Respuesta: *2*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *10.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *10.*

Respuesta *deepset/roberta-base-squad2*: *10.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *10.*

Respuesta *valhalla/t5-base-squad*: *10.*

B.2.3 Ejemplo de dificultad de selección del comienzo o final de la respuesta

The Alps are a crescent shaped geographic feature of central Europe that ranges in a 800 km (500 mi) arc from east to west and is 200 km (120 mi) in width. The mean height of the mountain peaks is 2.5 km (1.6 mi). The range stretches from the Mediterranean Sea north above the Po basin, extending through France from Grenoble, eastward through mid and southern Switzerland. The range continues toward Vienna in Austria, and east to the Adriatic Sea and into Slovenia. To the south it dips into northern Italy and to the north

extends to the south border of Bavaria in Germany. In areas like Chiasso, Switzerland, and Neuschwanstein, Bavaria, the demarcation between the mountain range and the flatlands are clear; in other places such as Geneva, the demarcation is less clear. The countries with the greatest alpine territory are Switzerland, France, Austria and Italy.

Pregunta: *How far does the Alps range stretch?*

Respuesta: *the Mediterranean Sea north above the Po basin, extending through France from Grenoble, eastward through mid and southern Switzerland*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *from the Mediterranean Sea north above the Po basin.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *The range stretches from the Mediterranean Sea north above the Po basin.*

Respuesta *deepset/roberta-base-squad2*: *800 km (500 mi).*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *from the Mediterranean Sea north above the Po basin,.*

Respuesta *valhalla/t5-base-squad*: *800 km (500 mi).*

B.2.4 Ejemplo de estructura sintáctica complicada

A substantial ***number of artists*** *and other figures have been* ***influenced*** *by, or* ***complimented****, West's work, including hip hop artists RZA of Wu-Tang Clan, Chuck D of Public Enemy, and DJ Premier of Gang Starr. Both Drake and Casey Veggies have acknowledged being influenced directly by West. Non-rap artists such as English singer-songwriters Adele and Lily Allen, New Zealand artist Lorde, rock band Arctic Monkeys, pop singer Halsey, Sergio Pizzorno of English rock band Kasabian and American indie rock group MGMT have cited West as an* ***influence****. Experimental and electronic artists such as James Blake*

Daniel Lopatin, and Tim Hecker have also cited West's work as an ***inspiration****. Experimental rock pioneer and Velvet Underground founder Lou Reed, in a review of West's album Yeezus, wrote that "the guy really, really, really is talented. He's really trying to raise the bar. No one's near doing what he's doing, it's not even on the same planet." Musicians such as Paul McCartney and Prince have also commended West's work. Famed Tesla Motors CEO and inventor Elon Musk complimented West in a piece for Time Magazine's 100 most* ***influential*** *people list, writing that:*

Pregunta: *A number of artists have cited Kanye as being what to them?*

Respuesta: *influential*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *influenced.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *influenced.*

Respuesta *deepset/roberta-base-squad2*: *inspiration.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *influence.*

Respuesta *valhalla/t5-base-squad*: *inspiration.*

B.2.5 Ejemplo de procesamiento de múltiples oraciones y razonamiento

Another class of knights were granted land by the prince, allowing them the economic ability to serve the prince militarily. A Polish nobleman living at the time prior to the 15th century was referred to as a "rycerz", very roughly equivalent to the English "knight," the critical difference being the status of "rycerz" was almost strictly hereditary; the class of all such individuals was known as the "rycerstwo". Representing the wealthier families of Poland and itinerant knights from abroad seeking their fortunes, this other class of rycerstwo, which became the szlachta/nobility ("szlachta" becomes the proper term for Polish nobility beginning about the 15th century), gradually formed apart from Mieszko I's and his

successors' elite retinues. This rycerstwo/nobility obtained more privileges granting them favored status. They were absolved from particular burdens and obligations under ducal law, resulting in the belief only rycerstwo (those combining military prowess with high/noble birth) could serve as officials in state administration.

Pregunta: *What positive did the szlachta class receive?*

Respuesta: *gradually formed apart from Mieszko I's and his successors' elite retinues.*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *more privileges granting them favored status.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *more privileges granting them favored status.*

Respuesta *deepset/roberta-base-squad2*: *more privileges granting them favored status.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *more privileges granting them favored status..*

Respuesta *valhalla/t5-base-squad*: *more privileges.*

B.2.6 Ejemplo de necesidad de conocimiento externo, inferencia y entendimiento

Additionally, there are around 60,000 non-Jewish African immigrants in Israel, some of whom have sought asylum. Most of the migrants are from communities in Sudan and Eritrea, particularly the Niger-Congo-speaking Nuba groups of the southern Nuba Mountains; some are illegal immigrants.

Pregunta: *Where are the non jewish immigrants from?*

Respuesta: *southern Nuba Mountains*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *Sudan and Eritrea.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *Sudan and Eritrea.*

Respuesta *deepset/roberta-base-squad2*: *Sudan and Eritrea,.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *Sudan and Eritrea.*

Respuesta *valhalla/t5-base-squad*: *Sudan and Eritrea.*

B.2.7 Ejemplo de tratado distinto o erróneo del lenguaje

According to Jan Nattier, the term Mahāyāna "Great Vehicle" was originally even an honorary synonym for Bodhisattvayāna "Bodhisattva Vehicle." The Aasāhasrikā Prajñāpāramitā Sūtra, an early and important Mahayana text, contains a simple and brief definition for the term bodhisattva: "Because he has enlightenment as his aim, a bodhisattva-mahāsattva is so called.".

Pregunta: *Where are the non jewish immigrants from?*

Respuesta: *Mahayana*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *Mahāyāna.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *Mahāyāna.*

Respuesta *deepset/roberta-base-squad2*: *Mahāyāna.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *Mahāyāna.*

Respuesta *valhalla/t5-base-squad*: *Mahyna.*

B.2.8 Ejemplo de dificultad de la respuesta por tipo o entidad

Raised in Chicago, West briefly attended art school before becoming known as a producer for Roc-A-Fella Records in the early 2000s, producing hit singles for artists such as Jay-Z

and Alicia Keys. Intent on pursuing a solo career as a rapper, ***West released*** *his debut album The College Dropout in* ***2004*** *to widespread commercial and critical success, and founded record label GOOD Music. He went on to explore a variety of different musical styles on subsequent albums that included the baroque-inflected Late Registration (2005), the arena-inspired Graduation (2007), and the starkly polarizing 808s & Heartbreak (2008). In 2010, he released his critically acclaimed fifth album, the maximalist My Beautiful Dark Twisted Fantasy, and the following year he collaborated with Jay-Z on the joint LP Watch the Throne (2011). West released his abrasive* ***sixth*** *album, Yeezus, to further critical praise in 2013. Following a series of recording delays and work on non-musical projects, West's* ***seventh*** *album, The Life of Pablo, was released in 2016.*

Pregunta: *How many total CDs has Kanye* ***West released*** *in his career so far?*

Respuesta: *7*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *2004.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *2004.*

Respuesta *deepset/roberta-base-squad2*: *sixth.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *2004.*

Respuesta *valhalla/t5-base-squad*: *seven.*

B.2.9 Ejemplo de dificultad de selección de la respuesta correcta entre varias

The ***Democratic Party*** *holds the majority of public offices. As of November 2008, 67% of registered voters in the city are* ***Democrats****. New York City has not been carried by a Republican in a statewide or presidential election since President Calvin Coolidge won the five boroughs in 1924. In 2012, Democrat Barack Obama became the first presidential*

candidate of any party to receive more than 80 % of the overall vote in New York City, sweeping all five boroughs. Party platforms center on affordable housing, education, and economic development, and labor politics are of importance in the city.

Pregunta: *Which political party holds the majority of most office terms in NYC?.*

Respuesta: *Democrats.*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *Democratic Party.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *Democratic.*

Respuesta *deepset/roberta-base-squad2*: *Democratic Party.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *Democratic Party.*

Respuesta *valhalla/t5-base-squad*: *Democratic.*

B.2.10 Ejemplo de fenómeno de complejidad y ambigüedades que hacen dudar también a una persona

Active solar techniques use photovoltaics, concentrated solar power, solar thermal collectors, pumps, and fans to convert sunlight into useful outputs. Passive solar techniques include selecting materials with favorable thermal properties, designing spaces that naturally circulate air, and referencing the position of a building to the Sun. Active solar technologies increase the supply of energy and are considered supply side technologies, while passive solar technologies reduce the need for alternate resources and are generally considered demand side technologies.

Pregunta: *What is an active solar technique used to generate energy?*

Respuesta: *designing spaces that naturally circulate air.*

Respuesta *phiyodr/bert-base-finetuned-squad2*: *photovoltaics.*

Respuesta *phiyodr/bert-large-finetuned-squad2*: *selecting materials with fa-*

vorable thermal properties.

Respuesta *deepset/roberta-base-squad2*: *concentrated solar power, solar thermal collectors, pumps, and fans.*

Respuesta *phiyodr/roberta-large-finetuned-squad2*: *concentrated solar power, solar thermal collectors, pumps, and fans.*

Respuesta *valhalla/t5-base-squad*: *photovoltaics.*

B.3 Errores cometidos por los modelos en base a la caracterización lingüística

BERT base	*BERT large*	*RoBERTa base*	*RoBERTa large*	*T5 base*	*SQuAD train*	*SQuAD dev*
Error	Error	Error	Error	Error	341 (0,39 %)	154 (2,60 %)
Error	Error	Error	Error	Ok	272 (0,31 %)	33 (0,56 %)
Error	Error	Error	Ok	Error	108 (0,12 %)	69 (1,16 %)
Error	Error	Error	Ok	Ok	63 (0,07 %)	28 (0,47 %)
Error	Error	Ok	Error	Error	15 (0,01 %)	22 (0,37 %)
Error	Error	Ok	Error	Ok	16 (0,01 %)	17 (0,29 %)
Error	Error	Ok	Ok	Error	25 (0,02 %)	27 (0,46 %)
Error	Error	Ok	Ok	Ok	53 (0,06 %)	51 (0,86 %)
Error	Ok	Error	Error	Error	255 (0,29 %)	22 (0,37 %)
Error	Ok	Error	Error	Ok	163 (0,18 %)	10 (0,17 %)
Error	Ok	Error	Ok	Error	310 (0,35 %)	25 (0,42 %)
Error	Ok	Error	Ok	Ok	382 (0,44 %)	28 (0,47 %)
Error	Ok	Ok	Error	Error	47 (0,05 %)	17 (0,29 %)
Error	Ok	Ok	Error	Ok	99 (0,11 %)	13 (0,22 %)
Error	Ok	Ok	Ok	Error	161 (0,18 %)	38 (0,64 %)
Error	Ok	Ok	Ok	Ok	794 (0,91 %)	154 (2,60 %)
Ok	Error	Error	Error	Error	48 (0,05 %)	13 (0,22 %)
Ok	Error	Error	Error	Ok	48 (0,05 %)	11 (0,19 %)
Ok	Error	Error	Ok	Error	34 (0,03 %)	17 (0,29 %)
Ok	Error	Error	Ok	Ok	38 (0,04 %)	23 (0,39 %)
Ok	Error	Ok	Error	Error	22 (0,02 %)	3 (0,05 %)
Ok	Error	Ok	Error	Ok	35 (0,04 %)	27 (0,46 %)
Ok	Error	Ok	Ok	Error	28 (0,03 %)	25 (0,42 %)
Ok	Error	Ok	Ok	Ok	124 (0,14 %)	68 (1,15 %)
Ok	Ok	Error	Error	Error	250 (0,28 %)	15 (0,25 %)
Ok	Ok	Error	Error	Ok	258 (0,29 %)	9 (0,15 %)
Ok	Ok	Error	Ok	Error	413 (0,47 %)	17 (0,29 %)
Ok	Ok	Error	Ok	Ok	1237 (10,42 %)	64 (1,08 %)
Ok	Ok	Ok	Error	Error	240 (0,27 %)	28 (0,47 %)
Ok	Ok	Ok	Error	Ok	2122 (20,44 %)	171 (2,88 %)
Ok	Ok	Ok	Ok	Error	1685 (10,94 %)	191 (3,22 %)
Ok	Ok	Ok	Ok	Ok	77135 (88,84 %)	4538 (76,55 %)

Tabla B.3: *Errores y aciertos en preguntas no null de modelos basados en BERT (phiyodr/bert-base-finetuned-squad-v2, phiyodr/bert-large-finetuned-squad-v2), RoBERTa (deepset/roberta-base-squad2, phiyodr/roberta-large-finetuned-squad2) y T5 (valhalla/t5-base-squad) para SQuAD*

B.3.1 Errores cometidos por clase de foco de la pregunta y tipo de entidad de respuesta

Clase	*BERT* squad *base*	*BERT* squad *large*	*RoBERTa* squad *base*	*RoBERTa* squad *large*	*T5* squad *base*
FNER-PERSON	9,19 % (25/272)	5,14 % (14/272)	4,77 % (13/272)	3,67 % (10/272)	9,55 % (26/272)
FNER-ORGANIZATION	10,00 % (1/10)	0,00 % (0/10)	0,00 % (0/10)	10,00 % (1/10)	0,00 % (0/10)
FNER-PERSON-ORG	9,96 % (26/261)	11,11 % (29/261)	6,51 % (17/261)	10,72 % (28/261)	11,11 % (29/261)
FNER-TITLE	100,00 % (1/1)	100,00 % (1/1)	100,00 % (1/1)	0,00 % (0/1)	0,00 % (0/1)
FNER-NATIONALITY	0,00 % (0/5)	0,00 % (0/5)	0,00 % (0/5)	0,00 % (0/5)	0,00 % (0/5)
FNER-RELIGION	0,00 % (0/9)	1,11 % (1/9)	1,11 % (1/9)	1,11 % (1/9)	1,11 % (1/9)
FNER-LOC	12,73 % (20/157)	10,19 % (16/157)	8,97 % (14/157)	12,10 % (19/157)	12,10 % (19/157)
FNER-COUNTRY	15,15 % (5/33)	6,06 % (2/33)	12,12 % (4/33)	6,06 % (2/33)	6,06 % (2/33)
FNER-LOCATION	0,00 % (0/2)	0,00 % (0/2)	0,00 % (0/2)	0,00 % (0/2)	0,00 % (0/2)
FNER-CITY	0,00 % (0/22)	0,00 % (0/22)	0,00 % (0/22)	0,00 % (0/22)	0,00 % (0/22)
FNER-DATE	0,00 % (0/5)	0,00 % (0/5)	0,00 % (0/5)	0,00 % (0/5)	0,00 % (0/5)
FNER-TIME	3,53 % (16/453)	1,98 % (9/453)	2,87 % (13/453)	3,31 % (15/453)	2,42 % (11/453)
FNER-NUMBER	5,08 % (15/295)	4,40 % (13/295)	4,06 % (12/295)	6,10 % (18/295)	4,40 % (13/295)
FNER-PERCENT	0,00 % (0/1)	0,00 % (0/1)	0,00 % (0/1)	0,00 % (0/1)	0,00 % (0/1)
Otras	10,84 % (182/1.678)	9,47 % (159/1.678)	9,53 % (160/1.678)	10,19 % (171/1.678)	11,91 % (200/1.678)
No aplica	4,81 % (417/8.669)	3,97 % (345/8.669)	3,49 % (303/8.669)	3,46 % (300/8.669)	4,39 % (381/8.669)

Tabla B.4: *Comparativa de errores en base a clase de foco de la pregunta y tipo de respuesta sobre preguntas con respuesta de SQuAD 2.0 dev*

Clase	*BERT* squad *base*	*BERT* squad *large*	*RoBERTa* squad *base*	*RoBERTa* squad *large*	*T5* squad *base*
FNER-PERSON	1,42 % (74/5.175)	0,48 % (25/5.175)	2,26 % (117/5.175)	1,93 % (100/5.175)	**4,09 % (212/5.175)**
FNER-ORGANIZATION	4,67 % (5/107)	2,80 % (3/107)	6,54 % (7/107)	11,21 % (12/107)	2,80 % (3/107)
FNER-PERSON-ORG	2,99 % (99/3.309)	1,26 % (42/3.309)	4,47 % (148/3.309)	4,44 % (147/3.309)	5,07 % (168/3.309)
FNER-TITLE	3,86 % (7/181)	2,76 % (5/181)	6,62 % (12/181)	6,62 % (12/181)	6,62 % (12/181)
FNER-NATIONALITY	0,00 % (0/62)	0,00 % (0/62)	1,61 % (1/62)	0 % (0/62)	1,61 % (1/62)
FNER-RELIGION	0,93 % (1/107)	0,93 % (1/107)	3,73 % (4/107)	5,61 % (6/107)	2,80 % (3/107)
FNER-IDEOLOGY	9,09 % (1/11)	0,00 % (0/11)	9,09 % (1/11)	9,09 % (1/11)	9,09 % (1/11)
FNER-LOC	2,36 % (55/2.328)	1,24 % (29/2.328)	4,59 % (107/2.328)	5,19 % (121/2.328)	3,17 % (74/2.328)
FNER-COUNTRY	4,26 % (20/469)	1,49 % (7/469)	5,33 % (25/469)	2,77 % (13/469)	4,26 % (20/469)
FNER-LOCATION	5,88 % (1/17)	5,88 % (1/17)	11,76 % (2/17)	5,88 % (1/17)	5,88 % (1/17)
FNER-CITY	1,61 % (5/309)	0,32 % (1/309)	1,69 % (5/309)	1,29 % (4/309)	2,26 % (7/309)
FNER-DATE	0,00 % (0/389)	0,00 % (0/389)	1,02 % (4/389)	0,00 % (0/389)	0,26 % (1/389)
FNER-TIME	1,24 % (100/8.004)	0,46 % (37/8.004)	1,46 % (117/8.004)	1,83 % (147/8.004)	1,43 % (115/8.004)
FNER-DURATION	50,00 % (1/2)	0,00 % (0/2)	0,00 % (0/2)	0,00 % (0/2)	50,00 % (1/2)
FNER-NUMBER	2,58 % (188/7.272)	1,32 % (96/7.272)	3,25 % (237/7.272)	2,40 % (175/7.272)	3,09 % (225/7.272)
FNER-PERCENT	7,69 % (1/13)	0,00 % (0/13)	0,00 % (0/13)	0,00 % (0/13)	0,00 % (0/13)
FNER-MONEY	0,00 % (0/1)	0,00 % (0/1)	0,00 % (0/1)	0,00 % (0/1)	0,00 % (0/1)
FNER-URL	0,00 % (0/2)	0,00 % (0/2)	50,00 % (1/2)	0,00 % (0/2)	0,00 % (0/2)
Otras	5,63 % (1.481/26.283)	1,75 % (460/26.283)	5,22 % (1.374/26.283)	5,97 % (1.570/26.283)	5,17 % (1.360/26.283)
No aplica	1,39 % (1.065/76.278)	0,73 % (563/76.278)	2,69 % (2.058/76.278)	2,52 % (1.923/76.278)	2,33 % (1.778/76.278)

Tabla B.5: *Comparativa de errores en base a clase de foco de la pregunta y tipo de respuesta sobre preguntas con respuesta de SQuAD 2.0 train*