UNAM iimas INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS

Embeddings contextualizados

Materia: Procesamiento de lenguaje natural

Blanca Vázquez

Embeddings contextuales

A diferencia de las estrategias anteriores, que generan un único vector para cada palabra independientemente del contexto, los embeddings contextuales generan diferentes vectores para la misma palabra dependiendo de su uso en la oración.

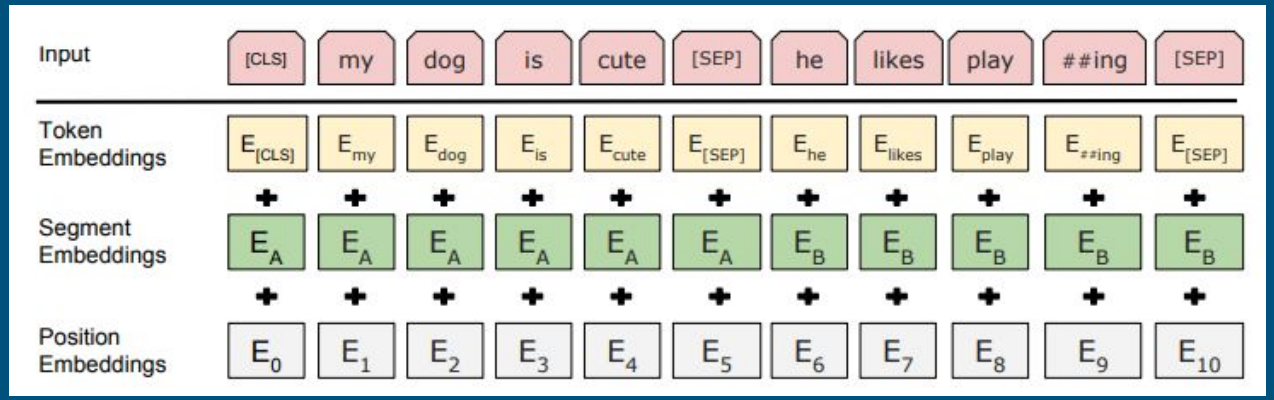
Ejemplo, la palabra "banco" tendrá un vector diferente en la oración

- "Me senté en el banco del parque" que en
- "El banco aprobó mi solicitud de crédito".

Esta variabilidad se logra entrenando el modelo con corpus de texto extensos de forma **bidireccional**, es decir, considerando no sólo las palabras que preceden a la palabra objetivo, sino también las que la siguen.

BERT (Bidirectional Encoder Representations from Transformers)

BERT es un modelo de código abierto propuesto por Devlin, Chang, Lee y Toutanova miembros de Google en el 2018. Está basado en mecanismos de atención y son bidireccionales.



Objetivo: general lenguaje

Imagen tomada de BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

BERT: corpus de entrenamiento

Wikipedia (2.5 B de palabras)	BooksCorpus de Google (Aproximadamente 800M de palabras)

- Adquirió conocimientos no solo en inglés, sino también en otros idiomas.
- Para el entrenamiento, Google desarrolló un nuevo hardware:
  - TPU (Unidad de Procesamiento Tensorial)

Versiones originales de BERT

Google entrenó dos versiones:

- BertBase:
  - 12 capas de transformación
  - 12 capas de atención
  - 110 millones de parámetros
- BertLarge:
  - 24 capas de transformación
  - 16 capas de atención
  - 340 millones de parámetros

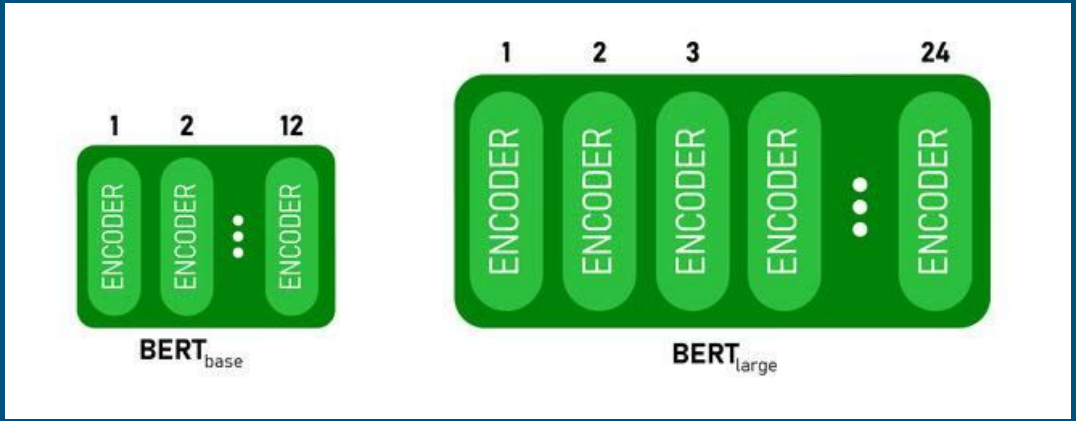


Imagen toma de https://www.geeksforgeeks.org/nlp/explanation-of-bert-model-nlp/

Diferencia entre Transformers y BERT

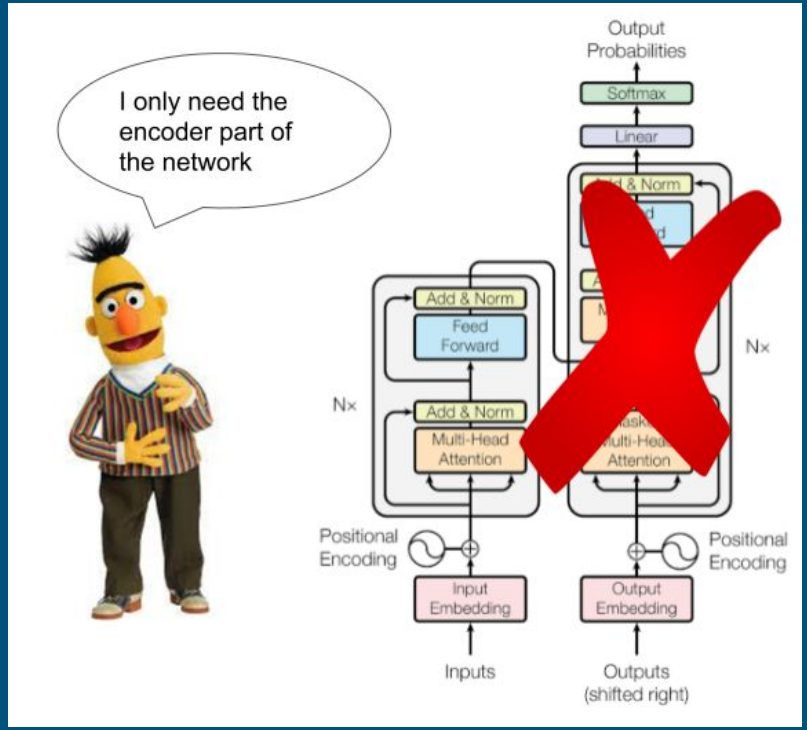


Imagen tomada de Cathal Horan, 2024.

¿Por qué usar únicamente el encoder?

- BERT presta un mayor énfasis en comprender las secuencias de entrada en lugar de generar secuencias de salida.
- Usando únicamente el encoder, BERT es capaz de codificar la información semántica y sintáctica en los embeddings.
- La salida final de BERT es un embedding, no es una salida textual.
- BERT es capaz de “observar” todas las palabras de una oración simultáneamente (bidireccional).

¿Cómo trabaja BERT?

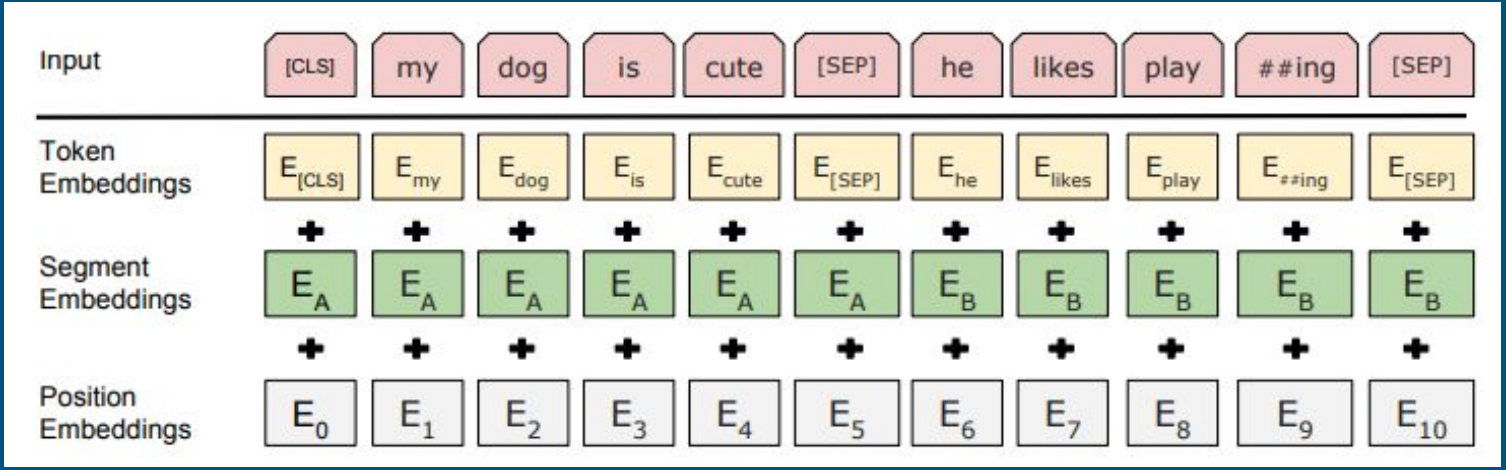


Imagen tomada de BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

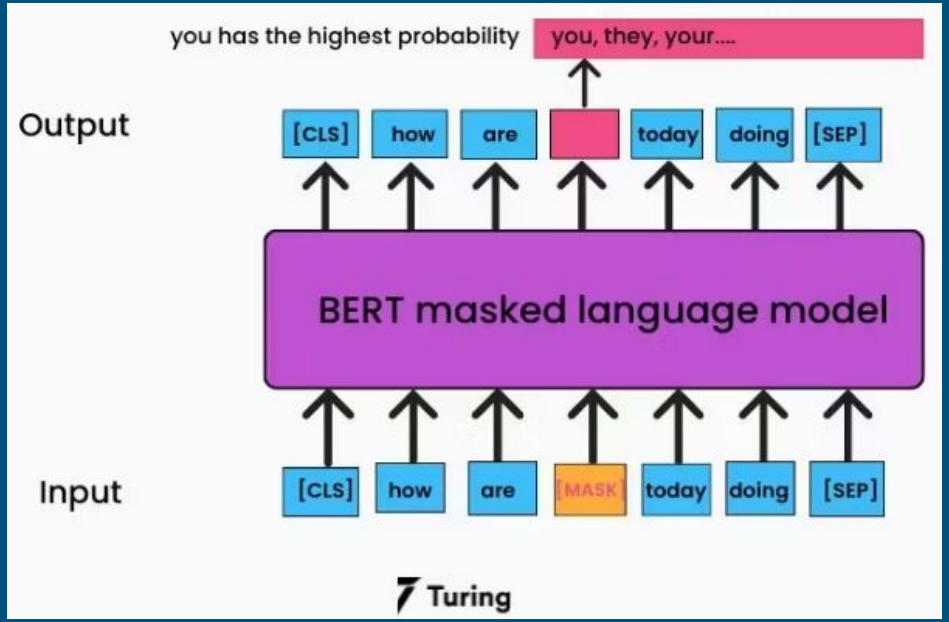
Mecanismos propuestos por BERT

- Masked Language Model (MLM)
  - En el proceso de pre-entrenamiento, se enmascara una parte de las palabras de cada secuencia de entrada y se entrena el modelo para predecir los valores originales de estas palabras enmascaradas según el contexto proporcionado por las palabras circundantes.

- Next Sentence Prediction (NSP)
  - En el proceso de entrenamiento, BERT aprende a comprender la relación entre pares de oraciones, prediciendo si la segunda oración sigue a la primera en el documento original.

Masked Language Model (MLM)

1. Se añade una capa de clasificación a la salida del codificador para predecir las palabras enmascaradas.
2. Los vectores de salida se multiplican por la matriz de embeddings, transformándolos en la dimensión del vocabulario. Esto para alinear las representaciones predichas con el espacio del vocabulario.
3. Para calcular la probabilidad de cada palabra, se usa una función SoftMax.
4. La función de pérdida utilizada durante el entrenamiento solo considera la predicción de los valores enmascarados.
5. El modelo se penaliza por la desviación entre sus predicciones y los valores reales de las palabras enmascaradas.

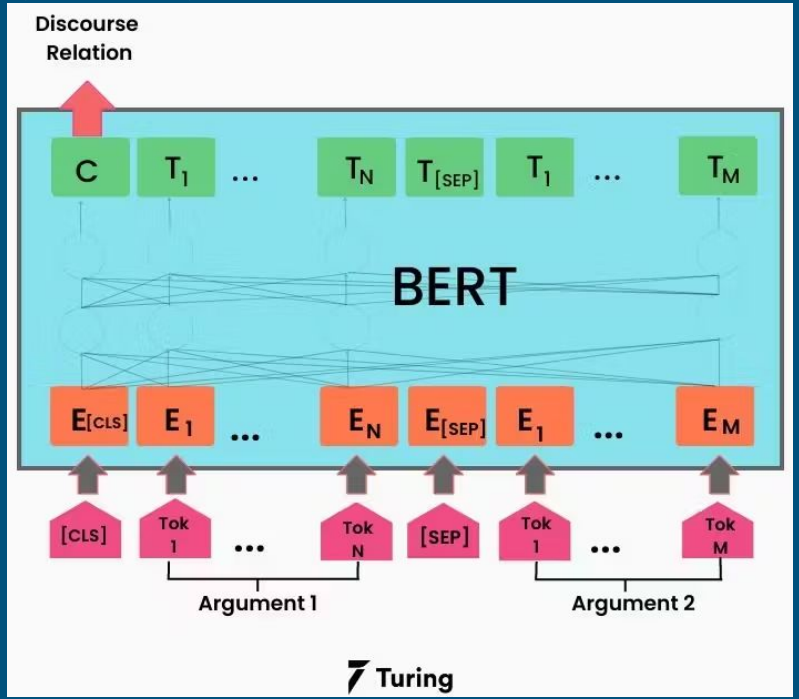


Durante el entrenamiento, una selección aleatoria del 15% de las palabras tokenizadas se enmascaran.

Imagen tomada de Turing, 2022.

Next Sentence Prediction (NSP)

1. Durante el entrenamiento, BERT predice si la segunda oración sigue a la primera en el documento original.
2. BERT transforma la salida del token [CLS] en un vector con forma de 2×1 mediante una capa de clasificación.
3. A continuación, calcula la probabilidad de que la segunda oración siga a la primera mediante SoftMax.



El 50% de los pares de entrada tienen la segunda oración como la oración subsiguiente en el documento original, y el otro 50 % tiene una oración elegida aleatoriamente.

Imagen tomada de Turing, 2022.

Importante

- Durante el entrenamiento, el MLM enmascarado y la predicción de la siguiente oración se entrenan conjuntamente.
- El modelo busca minimizar la función de pérdida combinada de ambos, lo que resulta en un modelo lingüístico robusto con mayor capacidad para comprender el contexto dentro de las oraciones y las relaciones entre ellas.

Aplicaciones de BERT

- Análisis de sentimientos
- Traducción de idiomas
- Respuesta a preguntas
- Búsqueda de Google
- Resumen de texto
- Coincidencia y recuperación de texto
- Resaltado de párrafos

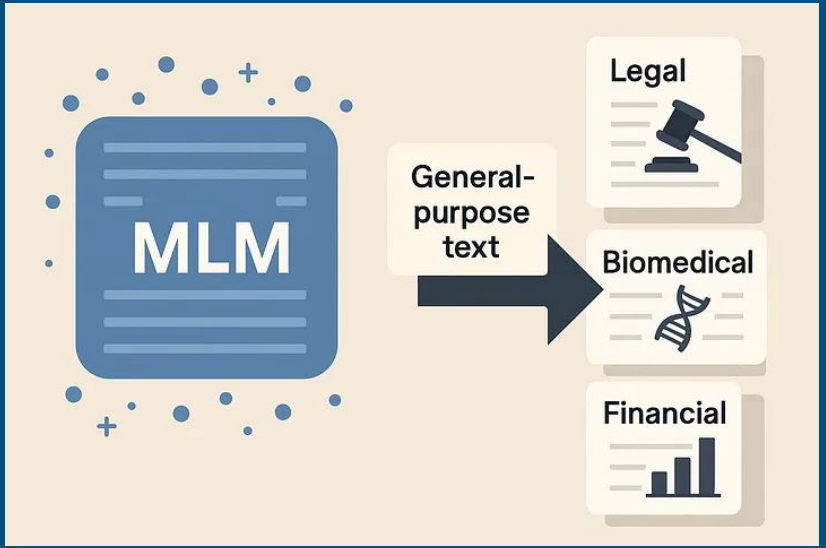


Imagen tomada de Okan Yenigün, 2025.

Variantes y adaptaciones del BERT

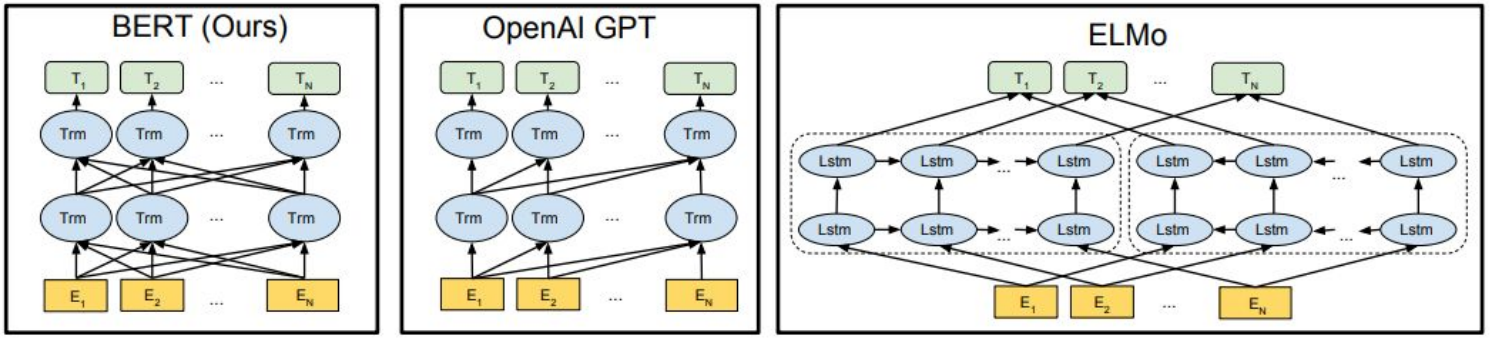


Figure 3: Differences in pre-training model architectures. BERT uses a bidirectional Transformer. OpenAI GPT uses a left-to-right Transformer. ELMo uses the concatenation of independently trained left-to-right and right-to-left LSTMs to generate features for downstream tasks. Among the three, only BERT representations are jointly conditioned on both left and right context in all layers. In addition to the architecture differences, BERT and OpenAI GPT are fine-tuning approaches, while ELMo is a feature-based approach.

Imagen tomada de BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
Para mayor información de ELMO, visita https://jalammar.github.io/illustrated-bert/

Variantes y adaptaciones del BERT

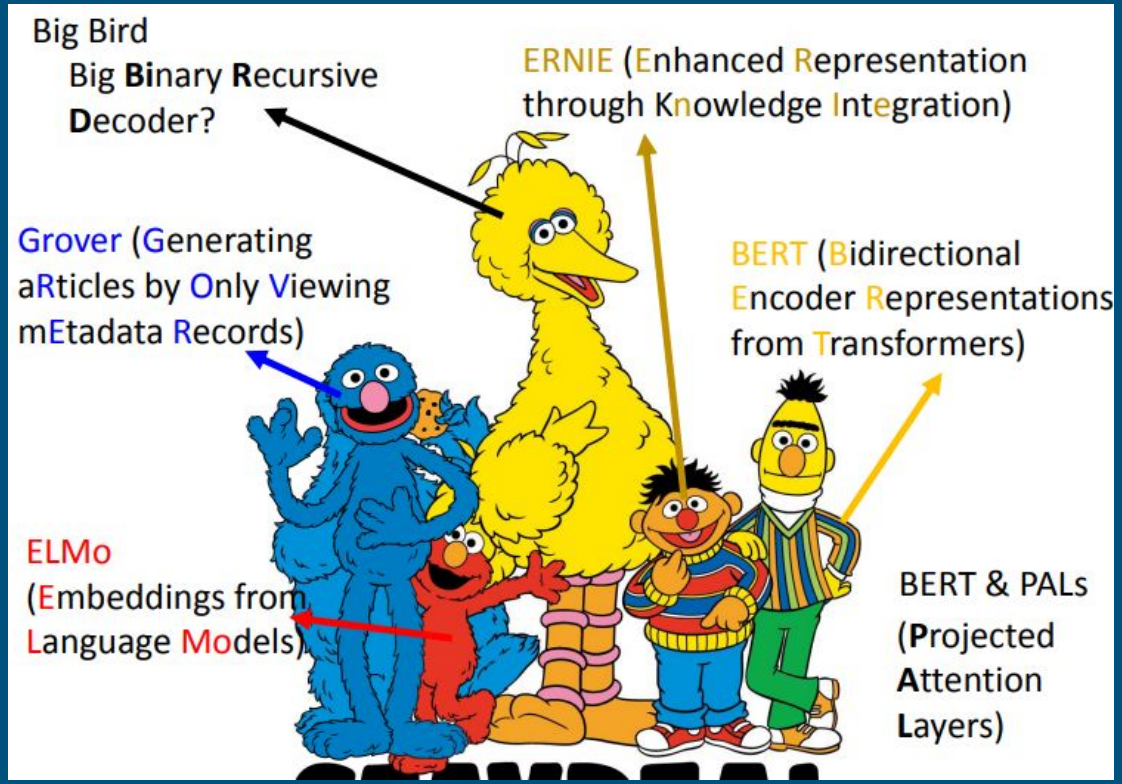


Imagen tomada de Hung-yi Lee

Variantes y adaptaciones del BERT

- ALBERT
- RoBERTa
- ELECTRA
- DistilBERT
- SpanBERT
- TinyBERT

- LegalBERT
- BioBert
- ClinicalBERT
- Protein Bert
- Berta
- …

Imagen tomada de Hung-yi Lee

Repositorios

- Google: https://jalammar.github.io/illustrated-bert/
- Hugging Face: https://huggingface.co/docs/transformers/model_doc/bert

GLUE score

GLUE por las siglas en inglés de *General Language Understanding Evaluation* es una colección de recursos para entrenar, evaluar y analizar sistemas de comprensión del lenguaje natural.

GLUE consta de nueve conjuntos de tareas “difíciles y diversas” diseñados para probar la comprensión del lenguaje de un modelo.

Acrónimo	Descripción
COLA	The Corpus of Linguistic Acceptability
SST-2	The Stanford Sentiment Treebank
MRPC	The Microsoft Research Paraphrase Corpus
QQP	The Quora Question Pairs
STS-B	The Semantic Textual Similarity Benchmark
MNLI	The Multi-Genre Natural Language Inference Corpus
QNLI	The Stanford Question Answering Dataset
RTE	The Recognizing Textual Entailment
WNLI	The Winograd Schema Challenge

La descripción detallada se encuentra disponible en: Glue: A Multi-Task Benchmark And Analysis Platform For Natural Language Understanding, 2019.

GLUE score

Corpus	|Train|	|Test|	Task	Metrics	Domain
			Single-Sentence Tasks		
CoLA	8.5k	**1k**	acceptability	Matthews corr.	misc.
SST-2	67k	1.8k	sentiment	acc.	movie reviews
			Similarity and Paraphrase Tasks		
MRPC	3.7k	1.7k	paraphrase	acc./F1	news
STS-B	7k	1.4k	sentence similarity	Pearson/Spearman corr.	misc.
QQP	364k	**391k**	paraphrase	acc./F1	social QA questions
			Inference Tasks		
MNLI	393k	**20k**	NLI	matched acc./mismatched acc.	misc.
QNLI	105k	5.4k	QA/NLI	acc.	Wikipedia
RTE	2.5k	3k	NLI	acc.	news, Wikipedia
WNLI	634	**146**	coreference/NLI	acc.	fiction books

Table 1: Task descriptions and statistics. All tasks are single sentence or sentence pair classification, except STS-B, which is a regression task. MNLI has three classes; all other classification tasks have two. Test sets shown in bold use labels that have never been made public in any form.

Imagen tomada de Glue: A Multi-Task Benchmark And Analysis Platform For Natural Language Understanding, 2019.

Model	Avg	Single Sentence CoLA	Single Sentence SST-2	Similarity and Paraphrase MRPC	Similarity and Paraphrase QQP	Similarity and Paraphrase STS-B	Natural Language Inference MNLI	Natural Language Inference QNLI	Natural Language Inference RTE	Natural Language Inference WNLI
Single-Task Training										
BiLSTM	63.9	15.7	85.9	69.3/79.4	81.7/61.4	66.0/62.8	70.3/70.8	75.7	52.8	**65.1**
+ELMo	66.4	**35.0**	90.2	69.0/80.8	85.7/65.6	64.0/60.2	72.9/73.4	71.7	50.1	**65.1**
+CoVe	64.0	14.5	88.5	73.4/81.4	83.3/59.4	67.2/64.1	64.5/64.8	75.4	53.5	**65.1**
+Attn	63.9	15.7	85.9	68.5/80.3	83.5/62.9	59.3/55.8	74.2/73.8	77.2	51.9	**65.1**
+Attn, ELMo	66.5	**35.0**	90.2	68.8/80.2	**86.5/66.1**	55.5/52.5	**76.9/76.7**	76.7	50.4	**65.1**
+Attn, CoVe	63.2	14.5	88.5	68.6/79.7	84.1/60.1	57.2/53.6	71.6/71.5	74.5	52.7	**65.1**
Multi-Task Training										
BiLSTM	64.2	11.6	82.8	74.3/81.8	84.2/62.5	70.3/67.8	65.4/66.1	74.6	57.4	**65.1**
+ELMo	67.7	32.1	89.3	**78.0/84.7**	82.6/61.1	67.2/67.9	70.3/67.8	75.5	57.4	**65.1**
+CoVe	62.9	18.5	81.9	71.5/78.7	84.9/60.6	64.4/62.7	65.4/65.7	70.8	52.7	**65.1**
+Attn	65.6	18.6	83.0	76.2/83.9	82.4/60.1	72.8/70.5	67.6/68.3	74.3	58.4	**65.1**
+Attn, ELMo	**70.0**	33.6	**90.4**	**78.0**/84.4	84.3/63.1	74.2/72.3	74.1/74.5	**79.8**	58.9	**65.1**
+Attn, CoVe	63.1	8.3	80.7	71.8/80.0	83.4/60.5	69.8/68.4	68.1/68.6	72.9	56.0	**65.1**
Pre-Trained Sentence Representation Models										
CBoW	58.9	0.0	80.0	73.4/81.5	79.1/51.4	61.2/58.7	56.0/56.4	72.1	54.1	**65.1**
Skip-Thought	61.3	0.0	81.8	71.7/80.8	82.2/56.4	71.8/69.7	62.9/62.8	72.9	53.1	**65.1**
InferSent	63.9	4.5	85.1	74.1/81.2	81.7/59.1	75.9/75.3	66.1/65.7	72.7	58.0	**65.1**
DisSent	62.0	4.9	83.7	74.1/81.7	82.6/59.5	66.1/64.8	58.7/59.1	73.9	56.4	**65.1**
GenSen	66.2	7.7	83.1	76.6/83.0	82.9/59.8	**79.3/79.2**	71.4/71.3	78.6	**59.2**	**65.1**

Table 4: Baseline performance on the GLUE task test sets. For MNLI, we report accuracy on the matched and mismatched test sets. For MRPC and Quora, we report accuracy and F1. For STS-B, we report Pearson and Spearman correlation. For CoLA, we report Matthews correlation. For all other tasks we report accuracy. All values are scaled by 100. A similar table is presented on the online platform.

Imagen tomada de Glue: A Multi-Task Benchmark And Analysis Platform For Natural Language Understanding, 2019.

Sitio oficial de GLUE

https://gluebenchmark.com/

Super GLUE

Nn nuevo punto de referencia inspirado en GLUE con un nuevo conjunto de tareas de comprensión del lenguaje más difíciles, recursos mejorados y una nueva tabla de clasificación pública.

https://super.gluebenchmark.com/

Super GLUE tareas

Acrónimo	Descripción
BoolQ	Boolean Questions
CB	CommitmentBank
COPA	Choice of Plausible Alternatives
MultiRC	Multi-Sentence Reading Comprehension
ReCoRD	Reading Comprehension with Commonsense Reasoning Dataset
RTE	Recognizing Textual Entailment
WiC	Word-in-Context
WSC	Winograd Schema Challenge

La descripción detallada se encuentra disponible en: SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems, 2019.

Super GLUE tareas

Table 1: The tasks included in SuperGLUE. *WSD* stands for word sense disambiguation, *NLI* is natural language inference, *coref.* is coreference resolution, and *QA* is question answering. For MultiRC, we list the number of total answers for 456/83/166 train/dev/test questions. The metrics for MultiRC are binary F1 on all answer-options and exact match.

Corpus	|Train|	|Dev|	|Test|	Task	Metrics	Text Sources
BoolQ	9427	3270	3245	QA	acc.	Google queries, Wikipedia
CB	250	57	250	NLI	acc./F1	various
COPA	400	100	500	QA	acc.	blogs, photography encyclopedia
MultiRC	5100	953	1800	QA	$F1_a$/EM	various
ReCoRD	101k	10k	10k	QA	F1/EM	news (CNN, Daily Mail)
RTE	2500	278	300	NLI	acc.	news, Wikipedia
WiC	6000	638	1400	WSD	acc.	WordNet, VerbNet, Wiktionary
WSC	554	104	146	coref.	acc.	fiction books

Imagen tomada de Glue: SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems, 2019.

Tabla de clasificación

Rank	Name	Model	URL	Score	CoLA	SST-2	MRPC	STS-B	QQP	MNLI-m	MNLI-mm	QNLI	RTE	WNLI
1	Microsoft Alexander v-team	Turing ULR v6		91.3	73.3	97.5	94.2/92.3	93.5/93.1	76.4/90.9	92.5	92.1	96.7	93.6	97.9
2	JDExplore d-team	Vega v1		91.3	73.8	97.9	94.5/92.6	93.5/93.1	76.7/91.1	92.1	91.9	96.7	92.4	97.9
3	Microsoft Alexander v-team	Turing NLR v5		91.2	72.6	97.6	93.8/91.7	93.7/93.3	76.4/91.1	92.6	92.4	97.9	94.1	95.9
4	DIRL Team	DeBERTa + CLEVER		91.1	74.7	97.6	93.3/91.1	93.4/93.1	76.5/91.0	92.1	91.8	96.7	93.2	96.6
5	ERNIE Team - Baidu	ERNIE		91.1	75.5	97.8	93.9/91.8	93.0/92.6	75.2/90.9	92.3	91.7	97.3	92.6	95.9

https://gluebenchmark.com/leaderboard/