Estrutura de validação de LLMs

"As consequências de a IA dar errado são graves, portanto, precisamos ser proativos em vez de reativos".
Elon Musk[94]

Marco

Os modelos de linguagem de grande escala (LLMs) oferecem um grande potencial para transformar vários setores e aplicativos, mas também apresentam riscos significativos que precisam ser abordados. Esses riscos incluem a geração de desinformação ou alucinações, a perpetuação de vieses, a dificuldade de esquecer informações aprendidas, preocupações éticas e de imparcialidade, problemas de privacidade devido ao uso indevido, dificuldades de interpretação dos resultados e a possível criação de conteúdo malicioso, entre outros.

Dado o impacto potencial desses riscos, os LLMs precisam ser completamente validados antes de serem implantados em ambientes de produção. De fato, a validação dos LLMs não é apenas uma prática recomendada, mas também um requisito regulatório em muitas jurisdições. Na Europa, o AI Act proposto exige a avaliação e a mitigação de riscos dos sistemas de IA[95], enquanto nos EUA, o framework de gestão de riscos de IA do NIST[96] e a AI Bill of Rights destacam a importância de entender e abordar os riscos inerentes a esses sistemas.

A validação dos LLMs pode se basear nos princípios estabelecidos na disciplina de risco de modelo, que se concentra[97] na avaliação e mitigação dos riscos decorrentes de erros, implementação inadequada ou uso indevido de modelos. Entretanto, no caso da AI e, particularmente, dos LLMs, é necessário adotar uma perspectiva mais ampla para englobar os outros riscos envolvidos. Uma abordagem abrangente da validação é essencial para garantir a implantação segura e responsável dos LLMs.

Essa abordagem holística está incorporada em uma estrutura de validação multidimensional para LLM, abrangendo aspectos fundamentais (Fig. 9), como risco de modelo, gestão de dados e privacidade, segurança cibernética, riscos legais e de compliance, riscos operacionais e tecnológicos, ética e reputação e risco de fornecedor, entre outros. Ao abordar todos esses aspectos de forma sistemática, as organizações podem identificar e mitigar proativamente os riscos associados aos LLMs, estabelecendo a base para aproveitar seu potencial de forma segura e responsável.

Nos LLMs, essa avaliação de risco pode ser ancorada nas seguintes dimensões usadas na disciplina de risco de modelo, adaptando os testes de acordo com a natureza e o uso do LLM:

- **Dados de entrada:** compreensão de texto[98], qualidade de dados[99].
- **Solidez conceitual e projeto do modelo:** seleção do modelo e de seus componentes (por exemplo, metodologias de fine-tuning, conexões de banco de dados, RAG[105]), e comparação com outros modelos[106].

[99]Elon Musk (nascido em 1971), CEO da X, SpaceX e Tesla. Empresário sul-africano, conhecido por fundar ou cofundar empresas como Tesla, SpaceX e PayPal, proprietário do X (antigo Twitter), uma rede social que tem seu próprio LLM, chamado Grok.

[100]European Parliament (2024) AI Act Art. 9: "Um sistema de gerenciamento de riscos deve ser estabelecido, implementado, documentado e mantido em relação aos sistemas de IA de alto risco. O sistema de gerenciamento de riscos [...] deverá [...] incluir [...] a estimativa e a avaliação dos riscos que podem surgir quando o sistema de IA de alto risco for usado de acordo com sua finalidade pretendida e em condições razoavelmente previsíveis de uso indevido".

[101]NIST (2023): "A decisão de comissionar ou implantar um sistema de IA deve ser baseada em uma avaliação contextual das características de confiabilidade e dos riscos, impactos, custos e benefícios relativos, e deve ser informada por um amplo conjunto de partes interessadas".

[102]Management Solutions (2014). Model Risk Management:Aspectos quantitativos e qualitativos.

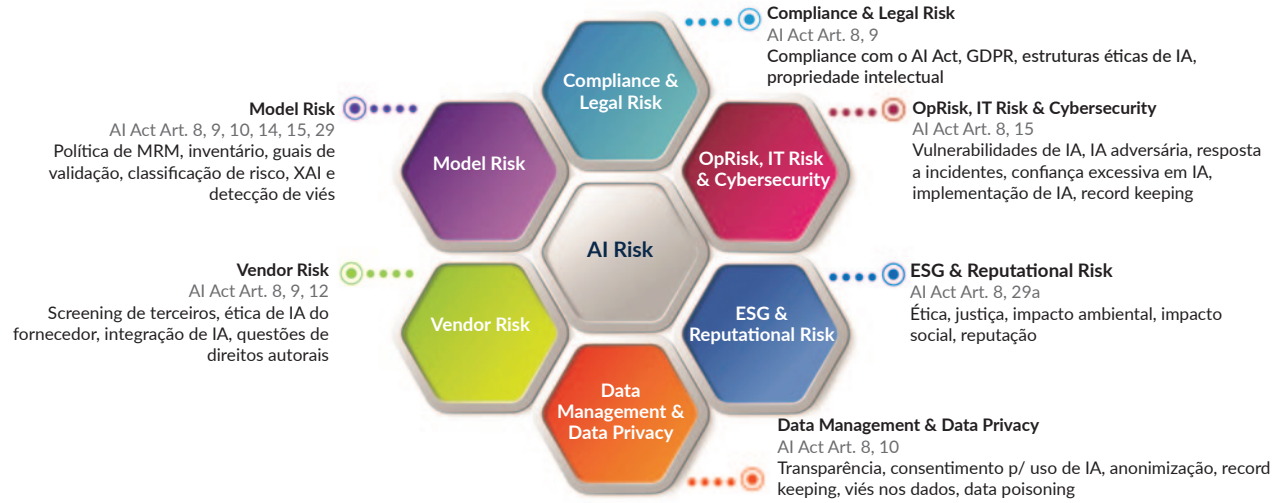
[103]Imperial et al. (2023).

[104]Wettig et al (2024).

[105]RAG (Retrieval-Augmented Generation) é uma técnica avançada na qual um modelo de linguagem busca informações relevantes de uma fonte externa antes de gerar o texto. Isso enriquece as respostas com conhecimento preciso e atual, combinando de forma inteligente a pesquisa de informações e a geração de texto. Ao integrar dados de fontes externas, os modelos RAG, como os modelos RAG-Token e RAG-Sequence propostos por Lewis et al. (2020), fornecem respostas mais informadas e consistentes, minimizando o risco de gerar conteúdo impreciso ou "alucinações". Esse avanço representa um passo significativo em direção a modelos de inteligência artificial mais confiáveis e baseados em evidências.

[106]Khang (2024).

Figura 9. Riscos associados à IA e referência regulatória na Lei de IA



- **Avaliação do modelo e análise de seus resultados:** privacidade e segurança dos resultados[107], precisão do modelo[108], consistência[109], robustez[110], adaptabilidade[111], interpretabilidade (XAI)[112], ética, vieses e imparcialidade[113], toxicidade[114], comparação com modelos desafiadores.

- **Implementação e uso:** revisão humana em uso (incluindo monitoramento de uso indevido), resolução de erros, escalabilidade e eficiência, aceitação do usuário.

- **Governança**[115] **e ética**[116]: estrutura de governança para IA generativa, incluindo LLM.

- **Documentação**[117]: integridade da documentação do modelo.

- **Compliance regulatório**[118]: avaliação dos requisitos regulatórios (por exemplo, AI Act).

Para garantir o uso eficaz e seguro dos modelos linguísticos, é essencial realizar uma avaliação de riscos que considere tanto o modelo em si quanto seu uso específico. Isso garante que, independentemente da origem (interna ou de um provedor) ou da personalização (fine-tuning), o modelo funcione adequadamente em seu contexto de uso, cumprindo as normas de segurança, éticas e regulamentares necessárias.

Técnicas de validação

Quando uma organização está pensando em implementar um LLM para um caso de uso específico, pode ser benéfico adotar uma abordagem holística que englobe as principais dimensões do ciclo de vida do modelo: dados, design, avaliação, implementação e uso. Além disso, de forma transversal, é necessário avaliar a conformidade com os regulamentos aplicáveis, como o AI Act na União Europeia.

Em cada uma dessas dimensões, dois conjuntos de técnicas complementares permitem uma validação mais completa (Fig. 10):

- Métricas de avaliação quantitativa (testes): são testes quantitativos padronizados que medem o desempenho do modelo em tarefas específicas. São benchmarks e métricas predefinidos para avaliar diferentes aspectos do desempenho do LLM após o pré-treinamento ou durante os estágios de fine-tuning ou instruction-tuning (ou seja, técnicas de aprendizagem por reforço), otimização, engenharia de prompts ou recuperação e geração de informações. Os exemplos incluem precisão na criação de resumos, robustez a ataques adversários ou consistência da resposta a solicitações semelhantes.

- Avaliação humana: envolve o julgamento qualitativo de especialistas e usuários finais, por exemplo, a análise de uma amostra específica de prompts e respostas do LLM por um ser humano para identificar erros.

A validação de um uso específico de um LLM é, portanto, realizada por meio de uma combinação de técnicas quantitativas (testes) e qualitativas (avaliação humana). Para cada caso de uso específico, é necessário projetar uma abordagem de validação sob medida, que consiste em uma seleção de algumas dessas técnicas.

[107] Nasr (2023).
[108] Liang (2023).
[109] Elazar (2021).
[110] Liu (2023).
[111] Dun (2024).
[112] Singh (2024).d
[113] NIST (2023). Oneto (2020), Zhou (2021).
[114] Shaikh (2023).
[115] Management Solutions (2014). Model Risk Management.
[116] Oneto (2020).
[117] NIST (2023).
[118] European Parliament (2024). AI Act.

Figura 10. Testes de avaliação de LLMs.

Dimensões	Aspectos validados	Descrição	Métricas e abordagens de validação (exemplos)	Avaliação humana (exemplos)
1. Dados de entrada	**1.1 Qualidade de dados**	Grau de qualidade da modelagem ou dos dados de aplicação	• Flesch-Kinkaid Grade	• Revisão caso a caso
2. Desenho do modelo	**2.1 Projeto do modelo**	Escolha de modelos e metodologias apropriados	• Revisão dos elementos de LLM: RAG, filtros de entrada ou saída, definição de prompts, fine-tuning, otimização... • Comparação com outros LLMs...	• Testes A/B
3. Avaliação do modelo	**3.1Privacidade e segurança**	Respeito à confidencialidade e não regurgitação de informações pessoais	• Data leakage • PII tests, K-anonymity	• Registros • *Hacking* ético
	3.2 Precisão	Correção e relevância das respostas do modelo	• Q&A: SummaQA, Word error rate • Recuperação de informações: SSA, nDCG • Resumo: ROUGE • Tradução: BLEU, Ruby, ROUGE-L • Outros: Sistemas de QA, nível de overrides, nível de alucinações... • Benchmarks: XSUM, LogiQA, WikiData	• Backtesting de forçagens • Revisão caso a caso
	3.3 Consistência	Respostas uniformes para consultas similares	• Cosine similarity measures • Jaccard similarity index	• Revisão caso a caso • Testes A/B
	3.4 Robustez	Resiliência a informações adversas ou enganosas	• Geração de texto adversarial (TextFooler), padrões Regex • Benchmarks de ataques adversários (PromptBench), número de refusals	• Hacking ético • Simulações de incidentes
	3.5.Adaptabilidade	Capacidade de aprender ou se adaptar a novos contextos	• Desempenho do LLM em novos dados por meio de Zero/One/Few-shot learning	• Testes A/B • Revisão caso a caso
	3.6 Explicabilidade	Compreensão do processo de tomada de decisão	• SHAP Scores de explicabilidade	• *Hacking* ético • *Focus groups*
	3.7 Vieses e imparcialidade	Respostas sem viés demográfico	• AI Fairness 360 toolkit • WEAT Score, paridade demográfica, word associations... • Benchmarks de vieses (BBQ...)	• *Hacking* ético • *Focus groups*
	3.8 Toxicidade	Propensão a geração de conteúdo nocivo	• Perspective API, Hatebase API • Toxicity benchmarks (RealToxicityPrompts, BOLD…)	• *Hacking* ético • *Focus groups*
4.Implementação e uso	**4.1 Revisão humana e segurança no uso**	Evite sugestões prejudiciais ou ilegais e inclua uma pessoa no circuito.	• Protocolos de risco, avaliações de segurança • Controle humano	• *Hacking ético* • *Focus groups*
	4.2 Recuperação e tratamento de erros	Capacidade de se recuperar de erros e lidar com entradas inesperadas	• Testes de recuperação do sistema • Métricas de processamento de erros	• Simulações de incidentes
	4.3 Escalabilidade	Manter o desempenho com mais dados ou usuários	• Stress testing do sistema, Apache Jmeter... • Benchmarks de escalabilidade	• Simulações de incidentes • Testes A/B
	4.4 Eficiência	Time-to-first-byte (TTFB), uso de GPU/CPU, inferência de emissões, memória, latência	• Time-to-first-byte (TTFB), uso de GPU/CPU, inferencia de emisiones, memoria, latencia	• Simulações de incidentes
	4.5 Aceitação do usuário	Teste de aceitação do usuário	• Checklist de requisitos do usuário, opt-out do usuário • Satisfação do usuário (Net Promoter Score, CSAT)	• Rastreamento de UX • Testes A/B

A seleção exata das técnicas dependerá das características específicas do caso de uso; em particular, há vários fatores importantes que devem ser levados em conta ao decidir sobre as técnicas mais adequadas:

- O nível de risco e a criticidade das tarefas a serem confiadas ao LLM.
- Se o LLM é aberto ao público (e, portanto, o hacking ético é de particular relevância) ou se seu uso é limitado ao uso interno da organização.
- Se a LLM processar dados pessoais.
- A linha de negócios ou serviço que o LLM usará.

Uma análise cuidadosa desses fatores permitirá a construção de uma estrutura de validação robusta adaptada às necessidades de cada uso de um LLM.

Métricas de avaliação quantitativa

Embora seja um campo de estudo emergente, existe uma ampla gama de métricas quantitativas para avaliar o desempenho do LLM. Algumas dessas métricas são adaptações daquelas usadas em modelos tradicionais de aprendizado de máquina, como precisão, exaustividade (recall), pontuação F1 ou área sob a curva ROC (AUC-ROC). Outras métricas foram projetadas especificamente para avaliar aspectos exclusivos dos LLMs, como a coerência do texto gerado, a fidelidade factual ou a diversidade de idiomas.

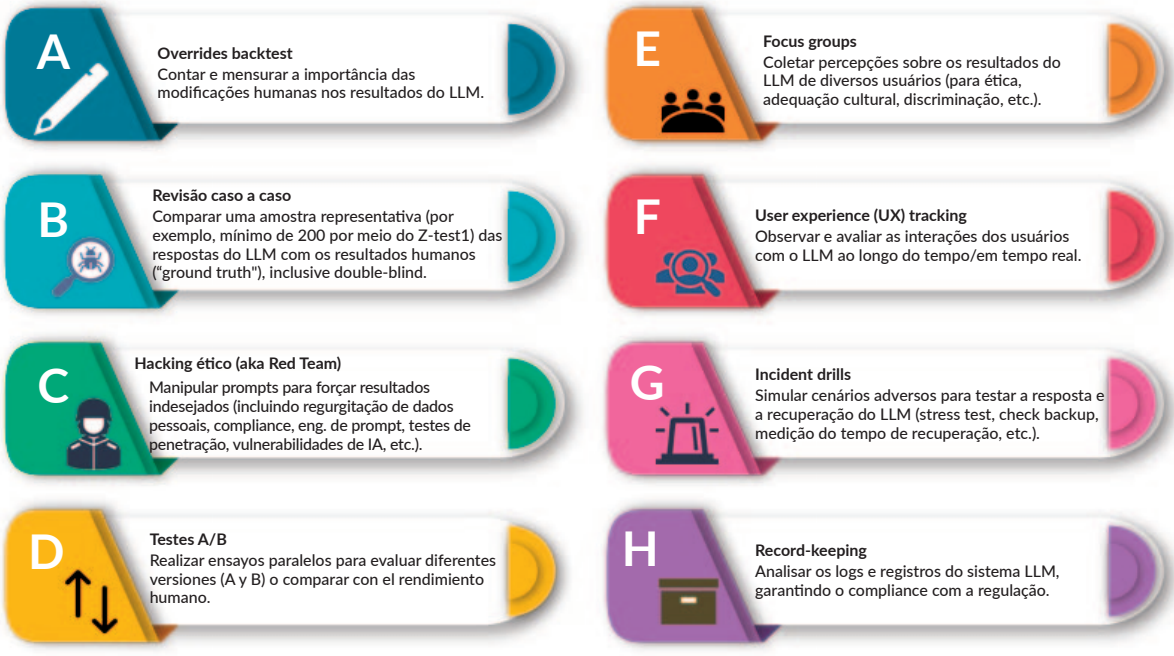
Nesse sentido, já existem estruturas holísticas de teste quantitativo de LLM em ambientes de programação Python, que facilitam a implementação de muitas das métricas de validação quantitativa; por exemplo:

- **LLM Comparator**[119]: uma ferramenta criada por pesquisadores do Google para a avaliação e comparação automática de LLMs, que analisa a qualidade das respostas dos LLMs.
- **HELM**[120]: avaliação holística de modelos de linguagem, que compila métricas de avaliação ao longo de sete dimensões (precisão, calibração, robustez, imparcialidade, viés, toxicidade e eficiência) para vários cenários predefinidos.
- **ReLM**[121]: sistema de validação e consulta de LLM usando o uso do idioma, incluindo avaliações de modelos linguísticos, memorização, viés, toxicidade e compreensão do idioma.

No momento, algumas técnicas de validação, como os métodos de explicabilidade baseados em SHAP (XAI), algumas métricas como ROUGE[122] ou análises de imparcialidade usando paridade demográfica, ainda não têm limites predefinidos amplamente aceitos. Nesses casos, cabe à comunidade científica e ao setor continuar a pesquisa para estabelecer critérios claros para uma validação robusta e padronizada.

[119] Kahng (2024).
[120] Liang (2023).
[121] Kuchnik (2023).
[122] Duan (2023).

Figura 11. Algumas técnicas de avaliação humana do LLM.



Técnicas de avaliação humana

Embora as métricas de avaliação quantitativa sejam mais diretamente implementáveis devido à grande quantidade de recursos e publicações on-line nos últimos anos, as técnicas de avaliação humana[123] são variadas e devem ser construídas de acordo com a tarefa específica[124] que está sendo executada pelo LLM, e incluem (Fig. 11):

- **Backtest do forçamento de usuários:** contabilizar e medir a importância das modificações humanas nos resultados do LLM (por exemplo, quantas vezes um gestor comercial precisa modificar manualmente os resumos de chamadas de clientes feitos por um LLM).
- **Revisão caso a caso:** comparar uma amostra representativa das respostas do LLM com as expectativas do usuário ("verdade básica").
- ***Hacking* ético *(Red Team)*:** manipulação de prompts para forçar o LLM a produzir resultados indesejados (por exemplo, regurgitação de informações pessoais, conteúdo ilegal, testes de penetração, exploração de vulnerabilidades).
- **Teste A/B:** comparação para avaliar duas versões do LLM (A e B), ou um LLM contra um ser humano.
- ***Focus groups:*** obtenção de feedback de vários usuários sobre o comportamento do LLM, por exemplo, sobre ética, adequação cultural, discriminação, etc.
- **Experiência do usuário *(UX tracking)*:** observar e avaliar as interações do usuário com o LLM ao longo do tempo ou em tempo real.
- **Simulações de incidentes:** simular cenários adversos para testar a resposta do LLM (por exemplo, teste de estresse, teste de backups, medição do tempo de recuperação, etc.).
- **Manutenção de registros:** analise os logs e registros do sistema do LLM, garantindo o compliance com os regulamentos e a trilha de auditoria.

[123]Datta, Dickerson (2023).
[124]Guzmán (2015).

Benchmarks de avaliação do LLM

A maioria dos modelos de inteligência artificial generativa, incluindo os LLMs, é testada usando benchmarks públicos que avaliam seu desempenho em uma variedade de tarefas relacionadas à compreensão e ao uso da linguagem natural. Esses testes servem para medir como o LLM lida com tarefas específicas e refletem a compreensão humana. Alguns desses benchmarks incluem:

- GLUE/SuperGLUE: avalia a compreensão do idioma por meio de tarefas que medem a capacidade do modelo de entender o texto.
- Eleuther AI Language Model Evaluation Harness: realiza uma avaliação "few-shot" dos modelos, ou seja, sua precisão com pouquíssimos exemplos de treinamento.
- ARC (AI2 Reasoning Challenge): testa a capacidade do modelo de responder a perguntas científicas que exigem raciocínio.
- HellaSwag: avalia o senso comum do modelo por meio de tarefas que exigem a previsão do final coerente de uma história.
- MMLU (Massive Multitask Language Understanding):): testa a precisão do modelo em uma ampla gama de tarefas para avaliar sua compreensão multitarefa.
- TruthfulQA: desafia o modelo a discernir entre informações verdadeiras e falsas, avaliando sua capacidade de lidar com dados verdadeiros.
- Winogrande: outra ferramenta de avaliação de senso comum, semelhante ao HellaSwag, mas com métodos e ênfase diferentes.
- GSM8K: avalia a capacidade lógico-matemática do modelo por meio de problemas matemáticos criados para os alunos.

Novas tendências

O campo da validação de LLMs está em constante evolução, impulsionado pelos rápidos avanços no desenvolvimento de modelos de LLM e pela crescente conscientização da importância de garantir sua confiabilidade, imparcialidade e alinhamento com a ética e a regulamentação.

A seguir estão algumas das principais tendências emergentes nesse campo:

- **Explicabilidade dos LLMs:** à medida que os LLMs se tornam mais complexos e opacos, há uma demanda crescente por mecanismos para entender e explicar seu funcionamento interno. As técnicas de XAI (eXplainable AI), como SHAP, LIME ou a atribuição de importância a tokens de entrada, estão ganhando destaque na validação de LLMs. Embora para os modelos tradicionais haja uma variedade de técnicas post-hoc disponíveis para entender o funcionamento dos modelos em nível local e global[125] (por exemplo, Anchors, PDP, ICE), e a definição e a implementação de modelos inerentemente interpretáveis por construção tenham proliferado, a implementação desses princípios para LLMs ainda não foi resolvida.

- **Uso de LLMs para explicar LLMs:** uma tendência emergente é usar um LLM para gerar explicações sobre o comportamento ou as respostas de outro LLM. Em outras palavras, um modelo de linguagem é usado para interpretar e comunicar de forma mais compreensível o raciocínio subjacente de outro modelo. Para enriquecer essas explicações, estão sendo desenvolvidas ferramentas[126] que também incorporam técnicas de análise post-hoc.

- **Técnicas de interpretabilidade *post-hoc***: essas técnicas baseiam-se na interpretabilidade dos resultados no estágio pós-treinamento ou de fine-tuning e permitem identificar quais partes da entrada influenciaram mais a resposta do modelo (importância de características), encontrar exemplos semelhantes no conjunto de dados de treinamento (similaridade baseada em embeddings) ou projetar prompts específicos que orientem o modelo para explicações mais informativas (estratégias de prompts).

- **Pontuações de atribuição:** como parte da interpretabilidade post-hoc, estão sendo desenvolvidas técnicas[127] para identificar quais partes do texto de entrada têm maior influência na resposta gerada por um LLM. Elas ajudam a entender quais palavras ou frases são mais importantes para o modelo. Há diferentes métodos para calcular essas pontuações:

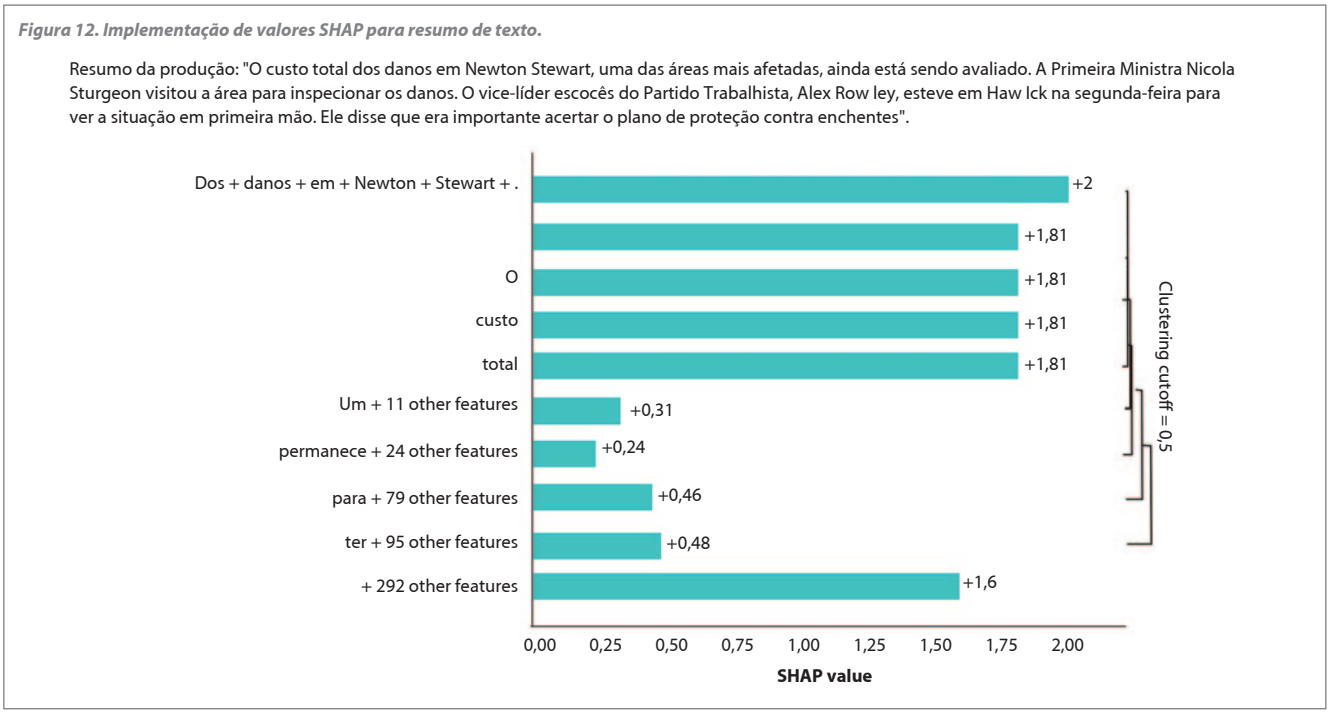
  - Métodos baseados em gradiente: analisam como os gradientes (uma medida de sensibilidade) mudam para cada palavra à medida que ela se move para trás na rede neural.

  - Métodos baseados em perturbações: modificam ligeiramente o texto de entrada e observam como a resposta do modelo muda.

  - Interpretação de métricas internas: eles usam métricas calculadas pelo próprio modelo, como pesos de atenção em transformers, para determinar a importância de cada palavra.

[125] Management Solutions (2023). Explainable Artificial Intelligence.
[126] Wang (2024).
[127] Sarti (2023).

Figura 12. Implementação de valores SHAP para resumo de texto.



Um exemplo de pontuação de atribuição é a aplicação da técnica SHAP para fornecer uma medida quantitativa da importância de cada palavra para o resultado do LLM, o que facilita sua interpretação e compreensão (Fig. 12).

- **Validação e monitoramento contínuos na produção:** além da avaliação oportuna antes da implantação, a prática de monitorar continuamente o comportamento dos LLMs quando eles estiverem em uso, como é feito com os modelos tradicionais, é bastante difundida. Isso possibilita a detecção de possíveis desvios ou degradações no desempenho ao longo do tempo, bem como a identificação de vieses ou riscos que não foram previstos inicialmente.

- **Validação colaborativa e participativa:** promove um maior envolvimento de diversas partes interessadas no processo de validação, incluindo não apenas especialistas técnicos, mas também usuários finais, órgãos reguladores, auditorias externas e representantes da sociedade civil. Essa participação pluralista permite a incorporação de diferentes perspectivas e promove a transparência e a responsabilidade.

- **Validação ética e alinhada à regulação:** além das métricas de desempenho, há um foco cada vez maior em avaliar se o comportamento do LLM é ético e alinhado aos valores humanos e à regulação. Isso envolve a análise de questões como imparcialidade, privacidade, segurança, transparência e o impacto social desses sistemas.

- **Machine unlearning:** essa é uma técnica emergente[128] que permite "desaprender" informações conhecidas de um LLM sem retreiná-lo novamente do zero. Isso é feito, por exemplo, adaptando os hiperparâmetros do modelo aos dados a serem desaprendidos. O mesmo princípio pode ser usado para remover as tendências identificadas. O resultado é um modelo que mantém seu conhecimento geral, mas que removeu as tendências problemáticas, melhorando sua imparcialidade e alinhamento ético de forma eficiente e seletiva. Vários métodos de machine unlearning estão sendo explorados atualmente, como o gradient ascent[129], o uso de fine-tuning[130] ou a modificação seletiva de determinados pesos, camadas ou neurônios do modelo[131].

[128] Liu (2024).
[129] Jang (2022).
[130] Yu (2023).
[131] Wu (2023)

SHAP (SHapley Additive exPlanations) aplicado a um LLM

O SHAP é um método de explicabilidade post-hoc baseado na teoria dos jogos cooperativos. Ele atribui a cada recurso (token) um valor de importância (valor Shapley) que representa sua contribuição para a previsão do modelo.

Formalmente, seja x = (x_1,…,x_n) uma sequência de tokens de entrada. A previsão do modelo é denotada como f(x). O valor de Shapley φ para o token x_i é definido como:

$$\phi_i = \sum_{\{S \subseteq N_i\}} \frac{\{|S|!\,(n - |S| - 1)!\}}{\{n!\}} [f(S \cup \{i\}) - f(S)]$$

em que N é o conjunto de todos os tokens, S é um subconjunto de tokens e f(S) é a previsão do modelo para o subconjunto

Intuitivamente, o valor de Shapley φi captura o impacto médio do token xi na previsão do modelo, considerando todos os subconjuntos possíveis de tokens.

Exemplo: Um LLM treinado para classificar e-mails corporativos como "importantes" ou "não importantes" é considerado. Dado o vetor de tokens de entrada:

X = [O, relatório, financeiro, do, Q2, mostra, um, aumento, significativo, na, receita e, na, rentabilidade].

O modelo classifica a correspondência como "importante" com f(x) = 0,84.

Aplicando o SHAP, são obtidos os seguintes valores de Shapley:

φ_1 = 0.01 (O)

φ_2 = 0.2 (relatório)

φ_3 = 0.15 (financeiro)

φ_4 = 0.02 (do)

φ_5 = 0.1 (Q2)

φ_6 = 0.05 (mostra)

φ_7 = 0.01 (um)

φ_8 = 0.15 (aumento)

φ_9 = 0.1 (significativo)

φ_{10} = 0.01 (na)

φ_{11} = 0.02 (receita)

φ_{12} = 0.12 (e)

φ_{13} = 0.01 (na)

φ_{14} = 0.02 (rentabilidade)

Interpretação: os tokens "relatório" (0,2), "financeiro" (0,15), "aumento" (0,15) e "receita" (0,12) têm as maiores contribuições para a classificação do e-mail como "importante". Isso sugere que o LLM aprendeu a associar esses termos à importância da mensagem em um contexto comercial.