

UNIVERSIDADE FEDERAL DO AMAZONAS  
ENGENHARIA ELÉTRICA COM ÊNFASE EM ELETRÔNICA

HELENA FARIAS CAVALCANTE

**SOBRE O CONSUMO DE ENERGIA EM DISPOSITIVOS ANDROID: UMA REVISÃO  
SISTEMÁTICA DA LITERATURA**

MANAUS, AMAZONAS  
2022

Helena Farias Cavalcante

Sobre o consumo de energia em dispositivos Android: uma revisão sistemática da literatura

Monografia apresentada à Coordenação do  
Curso de Engenharia Elétrica da  
Universidade Federal do Amazonas, como  
parte dos requisitos necessários à obtenção  
do título de Engenheiro Eletricista.

Aprovado em 09 de setembro de 2022.



---

Prof. Dr. Raimundo da Silva Barreto - Presidente e Orientador

Instituto da Computação - ICOMP - UFAM



---

Prof. Dra. Rosiane de Freitas Rodrigues - Membro

Instituto da Computação - ICOMP - UFAM



---

Prof. Dr. Celso Barbosa Carvalho - Membro

Departamento de Eletrônica e Computação - UFAM

## Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

C376s Cavalcante, Helena Farias  
Sobre o consumo de energia em dispositivos Android: uma  
revisão sistemática da literatura / Helena Farias Cavalcante, Edwin  
Juan Lopes Barboza Monteiro . 2022  
11 f.: il. color; 31 cm.

Orientador: Raimundo da Silva Barreto  
Orientadora: Rosiane de Freitas Rodrigues  
TCC de Graduação (Engenharia Elétrica - Eletrônica) -  
Universidade Federal do Amazonas.

1. Aplicativos Android. 2. Economia de Bateria. 3. Computação  
Móvel. 4. Consumo de Energia. 5. Eficiência Energética. I.  
Monteiro, Edwin Juan Lopes Barboza. II. Universidade Federal do  
Amazonas III. Título

# Sobre o consumo de energia em dispositivos Android: uma revisão sistemática da literatura

Helena Cavalcante  [ Universidade Federal do Amazonas | [helena.cavalcante@icomp.ufam.edu.br](mailto:helena.cavalcante@icomp.ufam.edu.br) ]

Edwin Monteiro  [ Universidade Federal do Amazonas | [edwin@icomp.ufam.edu.br](mailto:edwin@icomp.ufam.edu.br) ]

Raimundo Barreto  [ Universidade Federal Amazonas | [rbarreto@icomp.ufam.edu.br](mailto:rbarreto@icomp.ufam.edu.br) ]

Rosiane de Freitas  [ Universidade Federal do Amazonas | [rosiane@icomp.ufam.edu.br](mailto:rosiane@icomp.ufam.edu.br) ]

## Resumo

A redução do consumo de energia é um grande desafio na área de smartphones. Além do impacto ambiental causado pelo maior descarte de baterias degradadas, a experiência de uso é severamente impactada pelo rápido consumo de energia. Este trabalho apresenta uma Revisão Sistemática da Literatura (RSL), com foco em métricas de avaliação e técnicas para a redução do consumo de energia em smartphones Android, buscando identificar quais fatores impactam de maneira significativa na duração da bateria. A estratégia de busca resultou em 54 artigos, dos quais 14 foram identificados como relevantes para o estudo. Através da análise dessas publicações, foi possível perceber que o fator de maior impacto no consumo de bateria é o próprio comportamento do usuário, influência predominante no tempo ativo de uso e na preferência por aplicativos e configurações. Serviços de background e o modo de standby também são responsáveis por um alto consumo energético. Além disso, foi identificada uma tendência contínua e crescente em direção à análise dinâmica do perfil de uso dos usuários de smartphones Android, com especial relevância para estudos em grandes bases de dados criadas com informações coletadas de usuários em tempo de execução.

**Keywords:** *Aplicativos Android. Economia de Bateria. Computação Móvel. Consumo de Energia. Eficiência Energética. Smartphones.*

## 1 Introdução

Esta revisão sistemática tem enfoque em smartphones com sistema operacional Android por ser o sistema operacional amplamente utilizado em computação móvel e que tem ganhado destaque desde o seu lançamento pela Google e Open Handset Alliance em 2007. O Android é um sistema de código aberto baseado no kernel Linux e que permite a interação com usuários a partir de toques e gestos na tela do smartphone. Em estatísticas mais recentes levando em consideração o intervalo de 2009-2021 é possível notar pela Statcounter (2021) que o Android está presente em 39,75% dos smartphones em termos de mercado.

Apesar da popularidade em ascensão dos smartphones, novos desafios também são encontrados. O poder de processamento, assim como a complexidade dos aplicativos, evolui rapidamente a cada ano. Entretanto, apesar de uma maior capacidade de bateria, a autonomia energética não tem evoluído na mesma velocidade que os recursos inseridos ou aprimorados nos smartphones. A análise do consumo de energia em aplicativos é fundamental para a tomada de decisões buscando aumentar a duração de bateria, não apenas em termos de melhoria do conforto do usuário, mas também procurando reduzir os custos ambientais do descarte de baterias e de lixo eletrônico. O foco dessa revisão sistemática é o sistema operacional Android, visto que este é o que ocupa a maior fração do mercado, é um dos mais robustos e possui o código aberto com possibilidades maior de contribuição.

Em um estudo realizado por Pang et al. (2016), somente 18% em uma população de 122 desenvolvedores de aplicativos entrevistados consideraram a economia de energia durante o desenvolvimento e, somente 10% dos programadores

medem o consumo de energia de seus softwares. Esses baixos índices ocorrem, em grande parte, devido à falta de conhecimento dos programadores a respeito de técnicas, métodos, softwares e frameworks para a análise dessa importante métrica de desempenho de sistemas embarcados. Essa revisão da literatura na forma de mapeamento sistemático, busca sintetizar e avaliar as diversas abordagens na literatura atual, de modo a propiciar a pesquisadores, desenvolvedores e usuários uma visão compreensiva do estado da arte do consumo energético de *smartphones* Android. Para isso, foram considerados os resultados de publicações em língua inglesa nos últimos cinco anos que apresentem métricas para a avaliação ou técnicas para a redução do consumo de energia.

Este trabalho também identifica as principais técnicas e métodos usados na análise de consumo de energia e recomendações para desenvolvedores de aplicativo e usuários com a finalidade de reduzir o dreno de bateria de smartphones Android, a partir da identificação dos fatores de maior impacto na duração da bateria. A análise das publicações da literatura mostra uma clara tendência em direção a análises personalizadas do consumo de energia, considerando o impacto dos hábitos e opções do usuário. Vale destacar o interesse em recomendações personalizadas para cada perfil de usuário, com a finalidade de auxiliar nas ações a serem tomadas pelo usuário para economizar bateria.

Dessa forma, as questões de pesquisa que norteiam o desenvolvimento da presente revisão sistemática, elaboradas de acordo com as diretrizes fornecidas por Kitchenham and Charters (2007), são:

- **QP1:** “Quais técnicas e métodos são utilizados atualmente para investigar os fatores que impactam no con-

sumo de bateria de smartphones Android? ”;

- **QP2:** “Quais os fatores de maior impacto no consumo de bateria de smartphones Android? ”;
- **QP3:** “Quais as principais recomendações para reduzir o consumo de bateria de smartphones Android? ”.

As principais contribuições deste trabalho consistem em fornecer uma visão geral do estado da arte e identificar *gaps* na literatura sobre consumo de energia em smartphone Android. Através de uma visão global e compreensiva dos principais métodos, técnicas, ferramentas e experimentos descritos na literatura, este trabalho visa ser um guia: (i) aos pesquisadores, sugerindo novos e promissores caminhos de pesquisa; (ii) aos desenvolvedores, adicionando novas maneiras de criar aplicativos que drenem menos energia; e, (iii) aos usuários, indicando quais decisões de uso podem aumentar a duração da bateria de seus smartphones.

Este artigo está dividido da seguinte maneira: a Seção 2 detalha todos os pormenores do protocolo da RSL, descreve as estratégias de busca, as strings de busca, os critérios de inclusão e exclusão. A Seção 3 apresenta os resultados identificados na leitura e extração dos artigos avaliados. A discussão desses resultados é detalhada na Seção 4 e as perguntas de pesquisa são respondidas na Seção 5, e por fim, a Seção 6 discorre sobre as limitações dos trabalhos, lacunas para serem avaliadas e encerra com as considerações finais.

## 2 Metodologia

Esta seção descreve o protocolo da revisão sistemática realizada, cuja finalidade é estabelecer os fundamentos necessários para guiar a condução da revisão, a fim de identificar os principais métodos, técnicas, ferramentas e abordagens que lidam com consumo de energia em dispositivos Android.

### 2.1 Estratégias de busca

A consulta por artigos mediante uso de expressões - *strings* - de busca aconteceu na base de dados indexadora *Scopus*. Esta metabase agrega artigos de outras bases digitais como Elsevier, Springer e IEEE. Além da *Scopus*, as bases digitais da IEEE e da ACM também foram consultadas. Em termos de filtros, foram considerados apenas trabalhos publicados em inglês a partir de 2017 até o mês de outubro de 2021. A documentação e condução da Revisão Sistemática foram realizadas com o auxílio do software StArt (State of Art through Systematic Review), distribuído pela Universidade Federal de São Carlos (UFSCar, 2021). A Subseção 2.1 descreve a construção da string de busca desta revisão.

Foi aplicada a metodologia PIO (Population, Intervention, Outcomes), descrita por Kitchenham and Charters (2007) para a definição das strings de busca. Essa metodologia consiste em utilizar uma conjunção de disjunções de três blocos de palavras. Estes blocos contêm palavras e sinônimos associados, respectivamente, à população, intervenção e resultados. Os itens de cada aspecto da metodologia estão especificados na **Tabela 1**.

A definição das palavras e sinônimos presentes na composição da string surgiu de um processo iterativo dividido em

**Table 1.** Metodologia PIO (População, Intervenção e Resultados) aplicada no contexto desta RSL.

PIO	Descrição
<b>População</b>	Smartphones Android.
<b>Intervenção</b>	A partir de identificação do consumo, da carga e descarga e economia de bateria
<b>Resultados</b>	Técnicas, frameworks, bases de dados, métricas, arquiteturas e análises relacionadas ao consumo de energia em smartphones Android

duas etapas: (i) realização de uma pesquisa *ad-hoc* nas bases digitais da Scopus, ACM e IEEE baseada em termos chave extraídos dos artigos da lista de controle e; (ii) extração de termos feita nas bases digitais da IEEE, ACM e Scopus. As iterações foram controladas por meio do número de artigos relevantes retornados. Após essa etapa, uma pesquisa foi conduzida nestas mesmas bases por meio da string elaborada e apresentada na **Tabela 2**. No total, foram retornados 336 artigos, sendo 59 destes duplicados pelo fato da base Scopus indexar artigos da IEEE e ACM. Dos 336 artigos em análise, 246 foram obtidos da Scopus, 78 da IEEE, 6 da ACM e 6 inseridos manualmente.

### 2.2 Critérios de Inclusão e Exclusão

Os títulos, resumos e palavras-chave dos 277 trabalhos, resultantes da diferença entre os 336 estudos retornados pelas bases digitais e os 59 duplicados, foram sistematicamente lidos durante o primeiro filtro da etapa de seleção, para identificar os artigos relevantes. Os artigos de interesse deveriam abordar de maneira central o consumo de energia em smartphones Android, usar ou produzir bases de dados com métricas de energia relacionadas a população especificada, ou apresentar técnicas, métricas, abordagens, metodologias e recomendações voltadas para o tema da pesquisa.

Foram excluídos capítulos de livros, revisões da literatura, *surveys*, coletâneas de artigos, artigos duplicados e trabalhos cujo títulos não indicassem concordância com os objetivos da presente revisão sistemática. Artigos não disponíveis para leitura integral também foram excluídos. Também foram excluídos os artigos que lidavam com o consumo de energia, entretanto o foco de pesquisa não estava em smartphones. Alguns estudos, por exemplo, citavam intervenções para computadores automotivos e sistemas de automação residencial. Esses foram excluídos. Da mesma maneira, também foram excluídos trabalhos cujo foco era aplicativos web.

Um subconjunto significativo dos trabalhos apresentava intervenções ou melhorias em nível de programação. Tais artigos não foram considerados apropriados para o escopo desta revisão. Intervenções em nível de hardware foram igualmente excluídas, assim como trabalhos apresentando técnicas ou análises que não produzissem resultados em tempo de execução.

Para validar essa primeira etapa de seleção, feita por dois pesquisadores, foi calculado o critério Kappa de concordância entre pares. A metodologia detalhada por Perez (2020) serviu como base para o cálculo do coeficiente Kappa de Cohen. O refinamento dos critérios foi realizado em 3 iterações contendo 15 artigos para leitura, totalizando uma amostra

**Table 2.** Expressão de busca final compatível com as 3 bases de dados em estudo.

Base Indexadora	String	Data da busca
ACM, IEE e Scopus	( "android device*" OR "android smartphon*" OR "android app*" OR "android servic*" ) AND ( "battery charg*" OR "battery discharg*" OR "battery usage" OR "battery consumption" OR "energy consumption" OR "energy behavior" OR "energy efficiency" OR "battery powered" OR "energy saving" OR "saving energy" OR "power saving" OR "power consuming" ) AND ( approach* OR analy* OR metric OR measurement OR methodolog* OR model* OR framework* OR recommend* OR techniqu* OR dataset OR database )	01/05/2021

de 45 artigos. Os pesquisadores classificaram os artigos de maneira individual e, após cada iteração, fizeram uma reunião para ajuste dos critérios. A terceira e última iteração resultou em um coeficiente Kappa de 0.815, que representa, segundo Perez (2020), uma concordância quase perfeita entre os pesquisadores.

Um total de 54 artigos atendeu aos critérios de inclusão. Depois disso, foi realizada uma leitura mais direcionada, abrangendo as seções da introdução, metodologia e conclusão destes trabalhos durante a aplicação o segundo filtro da etapa de seleção. Dos 54 artigos, somente 24 continuaram a atender aos critérios de inclusão. Procedeu-se então à leitura integral e análise dos artigos na etapa de extração que ainda implicou na rejeição de 10 artigos, restando 14 artigos como estudos primários. A **Figura 1** mostra um esquemático do processo de seleção de artigos.

### 3 Sumarização dos Resultados

As publicações analisadas se encaixam dentro de quatro categorias, como mostra a **Tabela 3**: quatro artigos abordam o desenvolvimento de frameworks para a análise e/ou gerenciamento do consumo de energia, correspondendo a 28.57% do total de artigos; três trabalhos detalham as técnicas para aumento de durabilidade da bateria; outras três pesquisas abordam estudos de padrões com base no comportamento de usuários abrangendo 21,42% dos artigos; e por fim, quatro trabalhos lidam com o estudo controlado para determinar o padrão de consumo de energia. Entre os vários artigos analisados, há uma tendência nos experimentos que visa gerar as bases de dados ou ainda utilizar bases disponíveis com a finalidade de aplicar métricas voltadas para o consumo de energia por meio da mineração de dados. Duas dessas bases de dados, descritas em Pereira et al. (2021) e Rua et al. (2019), são abertas ao público em geral por escolha dos autores das pesquisas a elas associadas, visando o surgimento de novas linhas de pesquisa e avanço da comunidade científica na área de estudos de consumo de energia de dispositivos Android.

Os autores Guo et al. (2017) apresentam um estudo com dados de 80.000 usuários de smartphones, realizado através do aplicativo gerenciador de bateria Energy Saver, que coleta métricas relacionadas a maneira com que a bateria decai dependendo do uso dos aplicativos. É o primeiro estudo em larga escala dessa categoria. Até então, na literatura, só haviam sido realizados estudos de pequena ou média escala. A arquitetura do aplicativo tem dois componentes principais: o *EventListener* e o *EventServer*. O primeiro coleta dados todas as vezes que algum evento significativo ocorre no smart-

phone, como o decaimento da bateria, mudança no brilho da tela, abertura de um aplicativo, etc. O segundo transfere os dados coletados para um servidor remoto.

Após filtrar os dados, o programa calcula o tempo total de uso de cada aplicativo e o decaimento total de bateria durante o intervalo analisado. Com base nisso, calcula o consumo de energia de cada aplicativo. Os resultados foram validados usando um *setup* experimental com o software Monsoon Power Tool. Os resultados demonstram que há usuários com um padrão intenso de uso - em média 8.6h por dia - enquanto usuários com padrão leve de uso gastam menos de 1h. A média dos usuários no estudo tem um tempo de uso total de 3.79h por dia. Também foi descoberto que o estado de *standby* gasta bem mais energia do que o esperado, sendo responsável por 45% do consumo de energia total do celular em um dia.

Existem três abordagens possíveis para estimar o consumo de energia de um app. A primeira é por meio de instrumentação eletrônica. Essa abordagem retorna resultados mais precisos, porém requer hardware de alto custo e necessidade de conhecimento especializado. A segunda é por meio da construção de modelos de energia. Com maior facilidade de implementação, sua principal desvantagem reside na calibração dos parâmetros, nem sempre uma tarefa fácil. A terceira é por meio de software - simples de implementar, porém com baixa precisão nos resultados. O trabalho de Di Nucci et al. (2017) propõe a ferramenta PETrA, um framework para calcular o consumo de energia com a facilidade de implementação do software e a precisão de resultados do hardware. Sua metodologia é dividida em duas etapas: a simulação e o funcionamento da ferramenta.

Para a simulação, PETrA utiliza o programa *Android Monkey* para simular a interação do usuário com o aplicativo (cliques, toques, gestos, etc.). Para coletar os dados de consumo de bateria, PETrA utiliza ferramentas como *dmtrace-dump*, *Batterystats*, e *Systrace*, todos disponíveis pelo *Projeto Volta*. Para seu funcionamento, o framework depende da instalação de um aplicativo próprio no smartphone onde a coleta de dados será aplicada. Este aplicativo realiza todas as atividades de simulação e coleta, além de armazenar os resultados em um arquivo do tipo CSV. Por fim, o framework instalado no computador sumariza os resultados. Adicionalmente, o usuário pode visualizar os 5 métodos que mais consomem energia e ver o perfil de energia do aplicativo na forma de gráficos.

Para comparar a desempenho e determinar a acurácia da ferramenta, uma comparação foi realizada com a ferramenta Monsoon. O erro médio de estimativa foi de 4% em relação aos valores medidos pelo monitor Monsoon. 95% dos erros

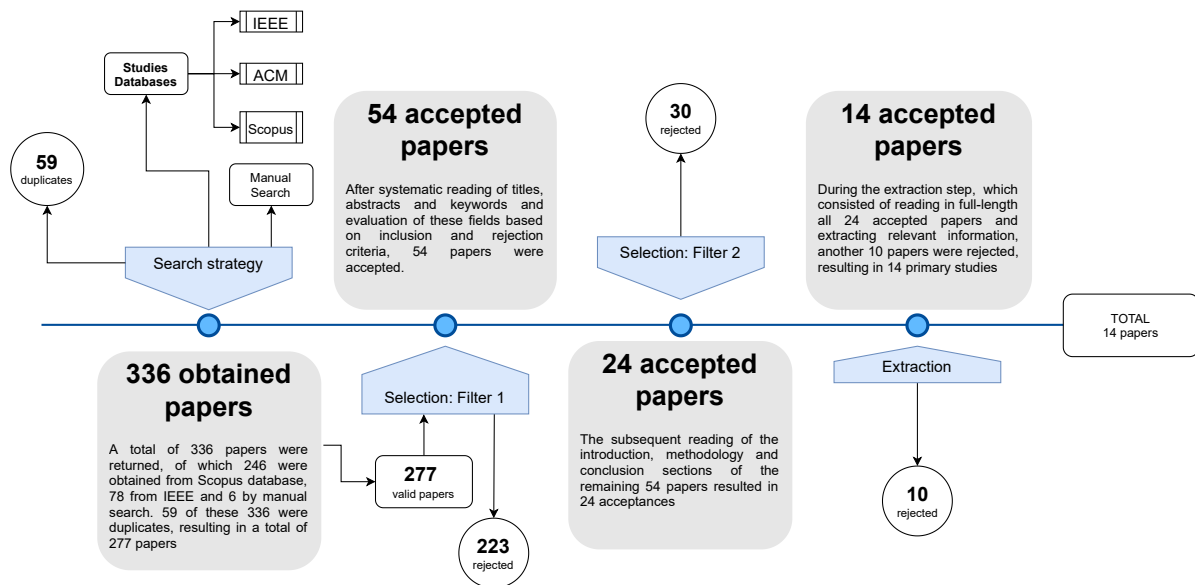


Figure 1. Número de artigos aceitos a cada etapa do processo de revisão sistemática.

Table 3. Classificação das publicações analisadas nesta revisão.

Tipo de Estudo	Estudos	Quantidade	Percentual
Estudo controlado para determinação de padrões de consumo de energia em aplicativos Android	Elliot et al. (2017)	4	28,57%
	Mehrotra et al. (2021)		
	Almasri and Sameh (2019)		
	Barreto Neto et al. (2020)		
Frameworks para análise e/ou gerenciamento do consumo de energia	Rua et al. (2019)	4	28,57%
	Di Nucci et al. (2017)		
	Wang et al. (2017)		
	Duan et al. (2017)		
Estudos de padrão de uso através de bases de dados	Guo et al. (2017)	3	21,42%
	Pereira et al. (2021)		
	Rua et al. (2019)		
Técnicas para redução de consumo de energia	Zhang et al. (2019)	3	21,42%
	Cañete et al. (2020)		
	Harihar and Sukumaran (2018)		
	Duan et al. (2017)		

ficaram a 5% dos valores reais.

Os autores Almasri and Sameh (2019) propõe um método para classificação de aplicativos quanto a seu consumo de energia em uma escala de 1 a 6 estrelas. Os autores apresentam uma maneira simples e intuitiva de mostrar ao público em geral, no ato da instalação, o impacto de um aplicativo na durabilidade da bateria. Eles realizam uma análise estática das permissões requeridas pelo aplicativo, coletadas por meio do arquivo AndroidManifest, e atrelam cada permissão a um determinado nível de consumo de energia com base em experimentos. Os autores aplicam seu sistema nas categorias top 5 do Google Play Store, que são: social, que incluem redes sociais como Facebook, Instagram e Twitter; ferramentas, aplicativos com funcionalidades diversas como o Google Play Service, Samsung My Files e o próprio app do Google; comunicações, categoria para aplicativos de chat como WhatsApp, Telegram e Viber; personalização, onde se encontram os aplicativos nos quais o usuário pode escolher novos papéis de parede, temas e ações para seus celulares; e, por fim, estilo de vida, que consiste em aplicativos para auxiliar no cotidiano do usuário, como assistentes virtuais (Alexa,

Siri), serviços de localização e integração com embarcados (Google Home, SmartThings), além de outros como Pinterest e Tinder. O trabalho conclui que a categoria que mais consome energia é a social.

Já Harihar and Sukumaran (2018) apresentam o Mobile Cloud Computing (MCC), um framework para seleção dinâmica de quais porções do código devem ser transferidas e executadas na nuvem. Para tanto, é necessário analisar se o custo de transferência dos dados de código durante o off-loading não consome mais bateria do que manter o código no dispositivo. Com esse fim, os autores buscam coletar os parâmetros de tempo de execução e transferência de dados.

Neste framework, todas as chamadas de métodos públicos são analisadas com relação ao tempo de execução e aos dados transferidos através de métodos de invocação. Para medir o custo do framework em termos de transferência de um determinado número de classes, o tamanho dos objetos de entrada de cada classe é computado usando as classes *ObjectOutputStream* e *ByteArrayOutputStream*. Esses cálculos são escritos em um arquivo binário, que é enviado à nuvem quando se detecta conexão com a internet. Os dados são en-

tão enviados a uma nuvem para mineração, e usadas para prever parâmetros de execução de classes, servindo como base para partição de código. Através de uma decomposição em séries temporais, os dados também são divididos em três categorias: (i) dados sazonais, que mostram variações cíclicas de dados, (ii) tendenciais, que mostram a tendência crescente ou decrescente dos dados, e (iii) residuais. Essa decomposição pode revelar padrões importantes de hábitos de uso.

Para testar o framework, foi criado um app customizado chamado *Go*, usado durante 30 dias por um número não informado de voluntários. Foi estimada uma redução no consumo de bateria de 3.5 por cento. Além disso, o trabalho demonstrou a superioridade de métodos dinâmicos em relação a métodos estáticos para estimar ganho de bateria. Foi possível também verificar a existência de padrões de comportamento dos usuários, mantendo uma carga extra de processamento devido a instrumentação de apenas 0.12%. Todo o processamento de MCC em si é feito na nuvem, economizando energia da bateria.

No trabalho descrito por Duan et al. (2017), é detalhada a implementação de um framework de gerenciamento de energia que, através de um algoritmo de aprendizagem não supervisionada, classifica os usuários em ativos e não ativos e gera uma decisão para aumentar a economia de bateria. A classificação se dá de acordo com o modo de uso predominante, baixo consumo ou alto consumo. A arquitetura geral tem três componentes: (i) o coletor de dados, que coleta as informações de bateria e também as informações de uso daquele usuário específico; (ii) o analisador de dados, que classifica os usuários usando um algoritmo de agrupamento; e (iii) o núcleo de decisão, que identifica padrões anormais de uso e notifica o usuário de quais ações ele pode tomar para melhorar o consumo de bateria. Para classificar os usuários, leva-se em conta o número de trocas da tela, duração da tela ligada em segundos, número de trocas entre segundo plano e primeiro plano, o volume de dados transferidos e o nível de bateria em porcentagem.

O sistema foi testado com 22 usuários e os autores concluíram que, de modo geral, o consumo de energia depende principalmente do tempo de uso ativo. Eles identificaram que existem duas causas principais para o consumo excessivo de bateria: comportamento anormal do usuário e ineficiência energética de dispositivos de modelos antigos. Assim, quando o núcleo de decisões identifica que um usuário não ativo está em modo de alto consumo, ele dá recomendações para diminuir o consumo.

Visando diminuir o impacto ambiental do descarte de baterias de celular, as quais não podem ser reutilizadas nem recicladas, Elliot et al. (2017) buscaram determinar e analisar experimentalmente a energia consumida por certos aplicativos multimídia e conduzir entrevistas para documentar o comportamento dos usuários e seu impacto no consumo de bateria. Utilizando o programa *Trepan*, foram coletadas informações da CPU e de temperatura. Os autores concluíram, com base nos experimentos, que o player de música da Samsung é mais econômico em relação ao player do Google; que o Youtube é o aplicativo de vídeo e streaming que mais consome bateria; e, que o aplicativo de mensagens de texto WhatsApp consome mais bateria que o Viber. Além disso, os autores encontraram diferença notável entre o dreno de bate-

ria do wi-fi e dos dados móveis. As entrevistas foram conduzidas com 3 usuários em seções de 15 a 20 minutos, nas quais foram feitas perguntas a respeito de seus hábitos de uso de smartphones. Entretanto, os achados das entrevistas não foram conclusivos na avaliação de Elliot et al. (2017).

O estudo produzido por Malavolta et al. (2020) apresenta o *Android Runner* (AR), um framework para facilitar a execução de experimentos com apps Android nativos e web apps. Android Runner é um framework que permite a replicabilidade e a customização de diferentes experimentos para análise de diferentes features e métricas em tempo de execução, como consumo de energia, uso da CPU e memória. O estudo para determinação da precisão do framework consistiu na criação de 27 apps open-source, cada qual com foco em um componente de software. Cada aplicativo rodou por 3 minutos, e cada rodada foi repetida 30 vezes com 2 minutos entre coletas sucessivas. A precisão estabelecida foi de 95%.

Outro framework, o *E-Spector*, é apresentado em Wang et al. (2017). Esse framework online de análise de consumo de energia é capaz de estimar o consumo em tempo de execução do aplicativo em teste, necessitando apenas do arquivo de instalação apk desse aplicativo. A ferramenta mostra todos os processos e serviços executados em qualquer momento da curva de energia do aplicativo. Para determinação da precisão do framework, um experimento foi feito com o *Monsoon Power Monitor*, no qual os resultados obtidos pelo monitor foram comparados aos resultados medidos pelo *E-Spector*. O *E-Spector* apresentou uma precisão de 10%, com overhead aproximado de 4%.

Já o estudo feito por Pereira et al. (2021) tem o objetivo de entender como aplicativos, sistemas operacionais, hardware e hábitos de uso influenciam no consumo de bateria. Os autores desenvolveram um aplicativo, *BatteryHub*, e disponibilizaram publicamente na loja de aplicativos oficial do Android. Os dados dos usuários do aplicativo foram periodicamente enviados a uma central, ajudando a compor uma base de dados com mais de 700 milhões de entradas de processos e dados relacionados a energia. Essa base de dados foi submetida a análises com métricas que revelaram tendências de características relacionadas ao consumo de energia, além de novas e promissoras linhas de pesquisa.

As tendências apontadas incluem: os smartphones em países subdesenvolvidos (Guiné Equatorial, Zimbábue, Botswana, Burundi, Benin) tendem a descarregar rapidamente e carregar devagar. Uma exceção foi a Groenlândia, onde os celulares carregam e descarregam lentamente. Uma possível causa apontada no trabalho foram as baixas temperaturas ambientes. Pereira et al. (2021) também encontraram que a diferença entre dispositivos de diferentes modelos é maior do que a diferença entre dispositivos de diferentes marcas e que, no geral, existe uma tendência de maior eficiência de energia para versões mais recentes do Android, atingindo o auge no sistema *Oreo*.

Aplicativos com serviços de localização e vídeo, como Google Maps, YouTube e VidMate têm tendências de consumo maior, enquanto aplicativos como Facebook, Messenger e Chrome apresentam tendências de consumo menores. Um achado do artigo foi a constatação de que o Facebook Lite consome mais bateria do que a versão principal do Face-



book. Ainda de acordo com Pereira et al. (2021), ativar ou desativar a localização e o Bluetooth não gera diferença, e o modo de economia de energia só é eficaz quando o celular está carregando. Os autores encontraram que a melhor conexão de rede é o Bluetooth Tethering, e que não há diferença entre Wi-fi e dados móveis. É importante notar que as métricas não indicaram um consumo exato de energia de cada amostra, aplicativo ou processo, somente tendências a serem investigadas.

Por sua vez, Barreto Neto et al. (2020) buscaram construir modelos de consumo de energia com base em padrões de uso do usuário, tendo em vista fornecer modelos mais precisos para serem usados por desenvolvedores e otimização automática. As características que mais impactam na bateria são escolhidas de acordo com um algoritmo de inteligência artificial. O framework *PiFA*, apresentado por Zhang et al. (2019), é um framework inteligente de ajuste de frequência de CPU, o qual regula a frequência da CPU de acordo com a fase de execução do aplicativo. *PiFA* consegue reduzir até 30% o consumo de energia de um aplicativo, sem incorrer em um overhead maior que 5%. Já Cañete et al. (2020) focam em construir mecanismos de adaptação que obtenham eficiência de energia ao adequar aplicativos Android aos padrões de uso do usuário.

A infraestrutura e base de dados *GreenHub*, desenvolvida por Rua et al. (2019), contém 609 aplicativos executáveis, um framework chamado *AnaDroid* para a automação de testes de consumo de energia a nível de código-fonte, e uma base de dados em larga escala com métricas relacionadas a energia dos aplicativos testados na ferramenta *AnaDroid*. A base de dados resultante possui 281.811 métricas estáticas e dinâmicas, das quais 39.375 são métricas de testes, 241.128 são métricas de métodos e 1.308 são métricas de classe.

Buscando determinar como variam os padrões de consumo de energia em relação aos aplicativos e respectivas configurações do sistema, Mehrotra et al. (2021) desenvolveram um algoritmo de classificação multiclasse que separa os aplicativos mobile em três categorias quanto ao consumo de energia: baixa, média e alta. O artigo detalha experimentos com diversos algoritmos de aprendizagem de máquinas para encontrar o modelo mais preciso. O objetivo da pesquisa foi investigar quais aplicativos consomem energia em excesso. Os autores encontraram que o consumo de energia do LCD é fator significativo para categorização de aplicativos, e que o melhor modelo de classificação é a floresta aleatória. Segundo o estudo, dados móveis consomem mais energia que o Wi-Fi e as funções notificação e sincronização não contribuem significativamente para o consumo de bateria. As categorias de entretenimento e propaganda apresentaram alto impacto na durabilidade da bateria.

Das publicações estudadas, o tipo predominante de análise é a análise dinâmica, presente em doze dos quatorze artigos. Por análise dinâmica, entende-se simulações, experimentos controlados ou coleta de dados por meio de aplicativo que visassem a análise do consumo de energia em tempo de execução, isto é, enquanto o usuário interage com o smartphone. Quatro artigos, Guo et al. (2017), Pereira et al. (2021), Dai et al. (2020) e Almasri and Sameh (2019), utilizam também análise estática, sendo que em dois Dai et al. (2020), Pereira et al. (2021), ela é feita em conjunto com a análise dinâmica.

As features coletadas mais comuns são status do Wi-Fi, de conexão de rede (3G/4G) e de Bluetooth, uso da CPU, taxas de transmissão em kBs e brilho da tela. Os números dessas features nos estudos primários são descritos na **Tabela 4** e na **Figura 3**. Para calcular o consumo da bateria e vincular esse consumo a um aplicativo, são coletadas, em geral, as informações dos aplicativos que estão em primeiro plano ou segundo plano, nível ou porcentagem da bateria e, em alguns casos, sua capacidade em mAh, e tensão e corrente da bateria.

Os principais softwares e bibliotecas para coleta de consumo de energia são o *Trepan Profiler*, *Android TrepanLib* e *PowerTutor*, *BatteryStats*, conforme mostrado na **Tabela 5**, na qual estão detalhados os números exatos. *Trepan Profiler* é um aplicativo Android para coleta de dados relacionados ao desempenho do smartphone, como frequência da GPU, energia consumida, memória e interfaces de rede, conforme informações de Qualcomm (2017). Já *PowerTutor* é um aplicativo Android para análise de consumo de energia, segundo Dick (2011). O *Batterystats* é uma ferramenta incluída no framework do Android que coleta dados da bateria do seu dispositivo (Google (2021)). Ele mostra a energia consumida por componentes principais de smartphones. Os principais softwares e bibliotecas para estudo comparativo de determinação de precisão em frameworks são o *BatteryStats Plugin* e, na maior parte dos estudos, o *Monsoon Power Monitor*, um dispositivo de hardware para medir, através da tensão e da corrente, a potência dissipada em smartphones. Alguns estudos também fazem estudos comparativos com instrumentação eletrônica própria. O principal software usado para realização de testes automatizados é o *Android Application Monkey*.

## 4 Discussão

Esta revisão da literatura identificou que as principais técnicas e métodos utilizados quando o tópico é analisar o impacto no consumo de bateria em smartphones estão divididos em: (i) coleta de informações exclusivamente da bateria como a tensão e/ou corrente – são métricas mais fáceis de se coletar e permitem obter outras métricas como a potência e a energia total consumida pelo dispositivo; (ii) captura de métricas por ferramentas que para esta finalidade são denominadas *profilers* tais como *Trepan*, *BatteryStats* e *PowerTutor* – fornecem capturas mais abrangentes como uso de CPU, nível de sinal, funcionalidades em uso como GPS, Bluetooth, dados móveis, volume de transferência de dados; e (iii) técnicas de Inteligência Artificial como árvores de decisão, florestas aleatórias, perceptron multicamadas e outros focadas na análise dos dados – geram modelos preditivos capazes de determinar o comportamento de uso do dispositivo e até reduzir o consumo de energia.

A partir do mapeamento de cada métrica foi possível elencar os fatores que têm maior impacto no consumo de energia. O perfil de uso do sistema apresenta-se como o maior agente causador do descarregamento de bateria, pois a depender de quais funcionalidades estão ativas, e por quanto tempo ficam em uso, podem ocasionar o maior consumo de energia. Um exemplo claro deste tipo de comportamento pode ser en-

contrado no trabalho de Duan et al. (2017) que classifica os usuários em ativos e não ativos obtendo resultados que reforçam a influência do tipo de usuário quanto ao consumo de energia além de inserir o fator qualidade/idade do hardware. Outro fator identificado por drenar grande parte da energia é a atividade em segundo plano, que é responsável por realizar atividades sem a necessidade interação com usuário, por exemplo, atualização de aplicativo, mecanismo de sincronização de mensagens e download de arquivos. Se o smartphone conter muitos aplicativos instalados, há uma probabilidade natural de muitas aplicações estarem em execução sem a consciência do usuário.

A identificação destes fatores ajuda a definir quais ações devem ser tomadas para evitar o elevado consumo de energia podendo ser algo simples de solucionar como a redução do nível de brilho da tela, ou ainda uma ação moderada como interromper algum aplicativo, ou em casos mais drásticos, efetuar a troca do aparelho.

Apesar dos trabalhos analisarem os dados gerados e/ou coletados, nenhum artigo forneceu uma recomendação direta do que deve ser feito para que haja a redução no consumo de energia de modo que a conclusão das análises remetem à indicativos do que pode ser levado em consideração para usuários de forma geral. Entre as limitações, pode-se citar que muitos trabalhos realizam experimentos em cenários controlados, nem sempre com a participação de usuários reais. Outros trabalhos até coletam dados reais, porém, a recomendação direcionada é mencionada apenas em pesquisas futuras.

Portanto, é possível enumerar diversas limitações aplicadas no escopo dos trabalhos e que essa revisão entende como uma lacuna potencial a ser investigada. Entre as limitações constam a baixa quantidade de informações e análises sobre os aplicativos em execução no segundo plano, experimentos controlados sem a presença de usuários reais, como ocorre na análise estática, além de algumas pesquisas que requerem, em muitos casos, o acesso ao código-fonte dos aplicativos, o que raramente ocorre devido ao código fechado das aplicações mais populares de empresas gigantes do ramo de mensagens instantâneas, streaming e jogos. Por fim, há a necessidade de não apenas gerar dados sobre a utilização do smartphone, mas também aplicar métodos adaptativos que forneçam recomendação personalizada conforme o padrão de utilização do usuário. Conforme apresentados em alguns trabalhos, o caminho está na coleta de features variadas do sistema a fim gerar bases suficientes para aplicar técnicas de aprendizagem de máquina que demonstraram ser promissoras e podem ser aplicadas na recomendação baseada em perfil comportamental do usuário a fim de fornecer uma recomendação mais precisa para evitar o alto consumo de energia do smartphone.

## 5 Respostas às questões de pesquisa

A. *Quais métodos e técnicas são utilizados atualmente para investigar os fatores que impactam no consumo de bateria de smartphones Android?*

Dos catorze artigos estudados, doze utilizam de forma direta o consumo de energia em um dado intervalo de tempo.

**Table 4.** Features mais coletadas nos artigos avaliados.

Features Coletadas	N. Artigos	Percentual
Aplicativos em foreground e background	14	100%
Nível da bateria	11	78,57%
Status da tela	11	78,57%
Wi-Fi	10	71,42%
Dados Móveis	10	71,42%
Tensão e/ou corrente da bateria	9	64,28%
CPU	8	57,14%
Taxas de transmissão	8	57,14%
Bluetooth	6	42,85%

Para saber esse consumo, três abordagens são utilizadas. A primeira é coleta de informações de tensão e/ou corrente da bateria, as quais são utilizadas para calcular a potência e, após obter a potência, a energia total consumida ao longo de um determinado período de tempo. Esse período é encontrado, em geral, pela subtração dos *timesteps*. As equações da potência (Equação 1) e de energia (Equação 2) são as mais usadas, nas formas aqui apresentadas ou em variações da mesma natureza. Um artigo se refere ainda à capacidade da bateria em mAh e à porcentagem de carga, enquanto outro calcula através de modelos automáticos baseados nas taxas de transmissão de rede, brilho da tela e uso da CPU.

$$P = V * I \quad (1)$$

$$E = \int P dt \quad (2)$$

A definição dos *timesteps* é variada. Em artigos como Barreto Neto et al. (2020), Guo et al. (2017), Almasri and Sameh (2019), Di Nucci et al. (2017), Harihar and Sukumaran (2018), ela é definida como o menor tempo em que não ocorre variação dos componentes do smartphone ou variações dessa regra. Desses cinco trabalhos a pesquisa de Barreto Neto et al. (2020) define um conjunto de regras para a definição de um período de tempo (*period slice*, no original): i) a amostra é do mesmo dispositivo, ii) as mudanças de bateria são no mesmo sentido, carregando ou descarregando e iii) sempre dentro de um limite de variação [-2, 2].

A segunda abordagem, utilizada em sete artigos, é coletar as informações de consumo de energia através de softwares especializados. Para esse fim, cinco usaram o software Trepp Profiler, e dois o software de gerenciamento de energia PowerTutor; também são citados o BatteryStats, Battery Linux profcs, GreenScaler e DroidWalker no total dos catorze artigos. Esses resultados são mostrados em detalhes na **Tabela 5**.

Há também um número significativo de técnicas estatísticas aplicadas à análise do consumo de bateria. Para a coleta e processamento de informações relacionadas à bateria a média simples é a mais utilizada. Almasri and Sameh (2019) e Dai et al. (2020) utilizam também cálculos de variância e desvio padrão para a generalização em larga escala dos traços de bateria. Os estudos que envolveram algoritmos de aprendizagem de máquina também fizeram análises estatísticas, sendo a principal delas a clusterização e agrupamento

**Table 5.** Identificação de softwares mais utilizados na coleta de dados para análise de consumo de energia.

Categoria do software	Softwares de coleta de consumo de energia	Estudos	Número	Percentual
Aplicativo Android de coleta de dados	Trepro Profiler	Rua et al. (2019) Cañete et al. (2020) Zhang et al. (2019) Malavolta et al. (2020) Elliot et al. (2017)	5	35,71%
Profilers diversos	Outros	Wang et al. (2017) Barreto Neto et al. (2020) Cañete et al. (2020) Zhang et al. (2019) Duan et al. (2017)	5	35,71%
Aplicativo Android de coleta de dados	PowerTutor	Almasri and Sameh (2019) Mehrotra et al. (2021)	2	14,28%
Ferramenta para coleta de dados por linha de comando	BatteryStats	Pereira et al. (2021) Malavolta et al. (2020)	2	14,28%

de amostras por similaridades, e clusterização por K-means. Barreto Neto et al. (2020) utiliza o Teste de Nemenyi. Pereira et al. (2021), para seleção de melhores algoritmos de modelos de consumo, usam plotagem de gráficos de caixa e matrizes de confusão. Os resultados estão apresentados em detalhes na **Figura 2**.

**Table 6.** Identificação da aplicação de técnicas de Aprendizagem de Máquina nos artigos avaliados.

Técnicas de IA	Percentual de Ocorrências
Clusterização	21,43%
Classificação Bayesiana	14,29%
Árvore de decisão	14,29%
LSTM	7,14%
Indução de regras	7,14%
Support Vector Regressor (SVR)	7,14%
Multilayer Perceptron Neural	7,14%

A respeito dos algoritmos de aprendizagem de máquina, foram utilizados diversos modelos, como mostra a **Tabela 6**. A técnica mais comum foi clusterização, presente em 21,43% dos trabalhos, seguida de classificação bayesiana e do algoritmo da árvore de decisão. Em Pereira et al. (2021), são utilizados os algoritmos do tipo J48 e Random Forest. Em Barreto Neto et al. (2020), é usado apenas o tipo Random Forest. Ainda nesse estudo, foi criado um algoritmo com o modelo Multilayer Perceptron Neural. Também foram usados o algoritmo de indução de regras e SVR (Support Vector Regressor). Os números exatos dos artigos estão mostrados na **Tabela 6** e na **Figura 4**.

**B. Quais são os fatores de maior impacto no consumo de bateria de smartphones Android?**

Oito dos artigos analisados realizaram estudo controlado ou análise de dados de usuários reais. A **Tabela 7** mostra os resultados obtidos. O fator de maior impacto encontrado nos estudos é o próprio perfil do usuário. Barreto Neto et al. (2020) realizou um experimento no qual foram reproduzidos dois vídeos diferentes no mesmo aparelho e nas mesmas condições. O resultado foi um consumo de energia diferente.

Duan et al. (2017) mostrou, também através de experimentos, que o tempo ativo de uso é um dos fatores principais para o aumento ou redução da vida da bateria. Esses experimentos caracterizam a relevância do comportamento do usuário no consumo de bateria.

Além disso, os aplicativos e serviços de background são citados como uma causa primária do dreno de bateria. Apesar da importância desse fator, apenas dois estudos fizeram estudos focados nele, o que pode revelar um *gap* na literatura. Elliot et al. (2017) realizou um experimento no qual demonstrou o alto consumo de energia dos serviços de background, rodando o mesmo aplicativo com quatro diferentes níveis de serviços ativos em segundo plano. O resultado do experimento foi que o nível com mais serviços ativos apresentou um consumo muito maior do que o nível com menos serviços ativos. Também são citados por como causas de dreno de bateria a ineficiência de hardwares antigos em Guo et al. (2017), Di Nucci et al. (2017); e o modo de standby. Em um experimento realizado por Dai et al. (2020), foi mostrado que após o desligamento do celular, há um *delay* de cerca de seis segundos até a diminuição do nível de corrente, explicitando a continuidade dos serviços de background por algum tempo após o smartphone entrar em modo de standby. Além disso, ainda segundo Dai et al. (2020), alguns aplicativos mantêm o smartphone ligado, mesmo que a tela esteja desligada, drenando a bateria.

**Table 7.** Sumarização dos fatores que afetam o desempenho dos dispositivos e sua ocorrência nos trabalhos analisados.

Fatores de maior impacto	Percentual de ocorrência
Perfil de usuário	28,57%
Rede móvel e Wifi	21,43%
Aplicativos e serviços em segundo plano	14,29%
Hardware antigo	14,29%
Modo standby	14,29%

**C. Quais são as principais recomendações para reduzir o consumo de bateria de smartphones Android?**

Apenas dois artigos fornecem recomendações diretas so-

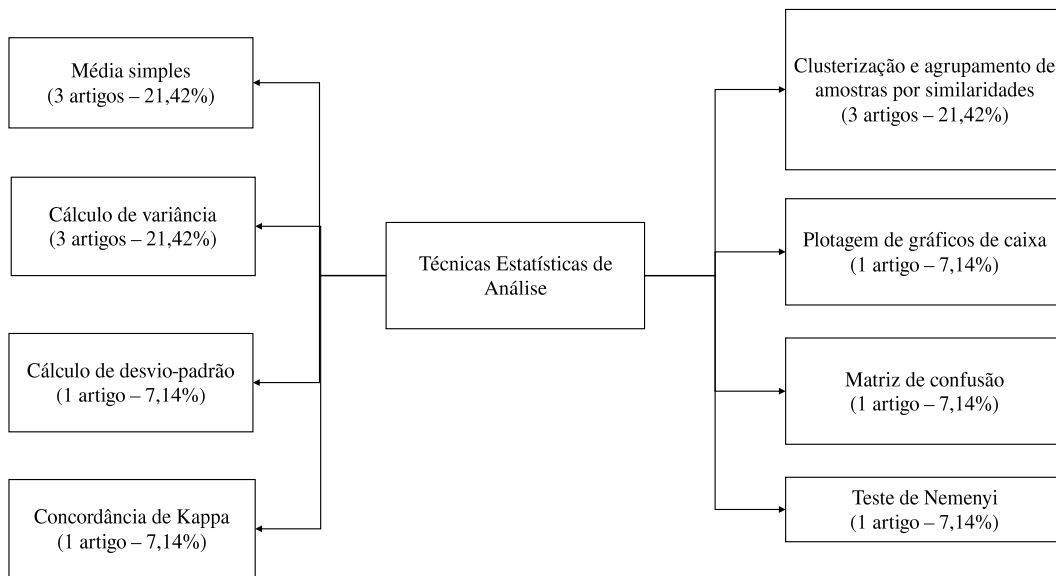


Figure 2. Técnicas estatísticas de análise mais utilizadas atualmente.

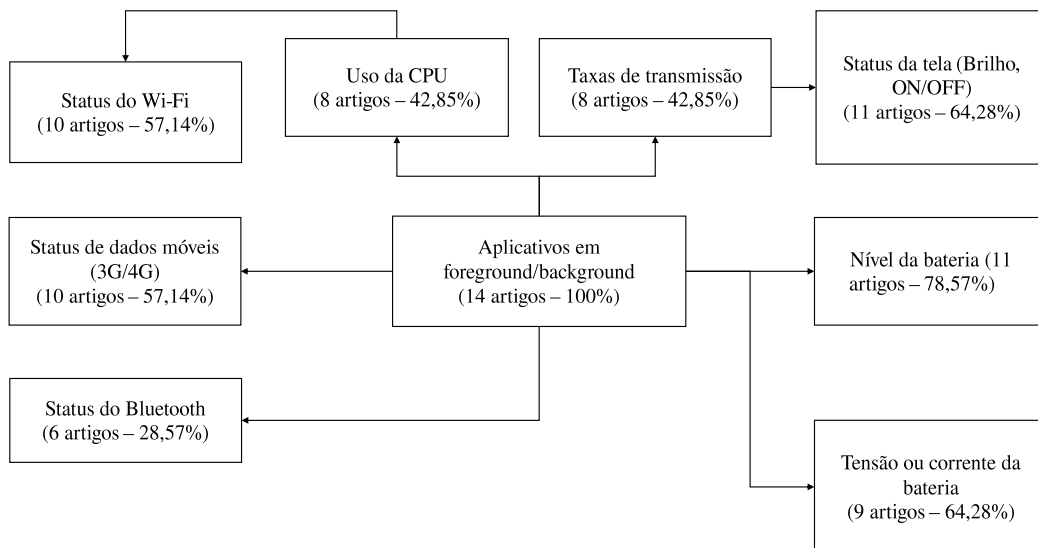


Figure 3. Features do dispositivo e do sistema mais comumente coletadas para análise do consumo de energia.

bre como reduzir o consumo de bateria de smartphones Android. Os demais estudos, embora tenham realizado análises sobre os fatores de maior impacto, não fornecem recomendações diretas ou limitam-se apenas em identificar o fator de maior consumo de energia. Entre as recomendações identificadas, destaca-se: (i) a investigação e otimização das atividades de background durante o modo de standby; (ii) aos pesquisadores, recomenda-se manter o foco de pesquisa de energia na pequena fração dos aplicativos existentes usados com frequência e que consomem mais energia; (iii) para usuários, pode-se desligar interfaces de rede não usadas, diminuir o brilho da tela e interromper a execução de serviços de background quando o dispositivo estiver em modo standby. Outra sugestão é não utilizar smartphones com hardware antigo e ineficiente. Para desenvolvedores, recomenda-se atualizar a plataforma HW/SW, evitar aplicativos e plataformas de hardware pouco eficientes e otimizar

as configurações do sistema.

Além disso, é recomendado que se busque comunicar melhor aos usuários como preservar suas baterias e proteger o meio ambiente. Usuários conscientes esperam baterias que durem mais tempo e smartphones que consumam menos energia, pressionando desenvolvedores a pensar nessas questões. Além disso, recomenda-se aos desenvolvedores que utilizem o Green Code para diminuir o consumo de bateria. Recomenda-se também que estudos sejam feitos no sentido de manter a evolução das tecnologias de bateria sincronizadas com a evolução dos smartphones.

Entre as recomendações que se pode inferir através dos resultados dos artigos, cita-se adquirir modelos de celulares de marcas com maior eficiência de energia, buscar manter a versão do Android sempre atualizada ou, de preferência, usar o Android Oreo, e reduzir o tempo de uso de aplicativos com funcionalidades multimídia, como jogos e apps de

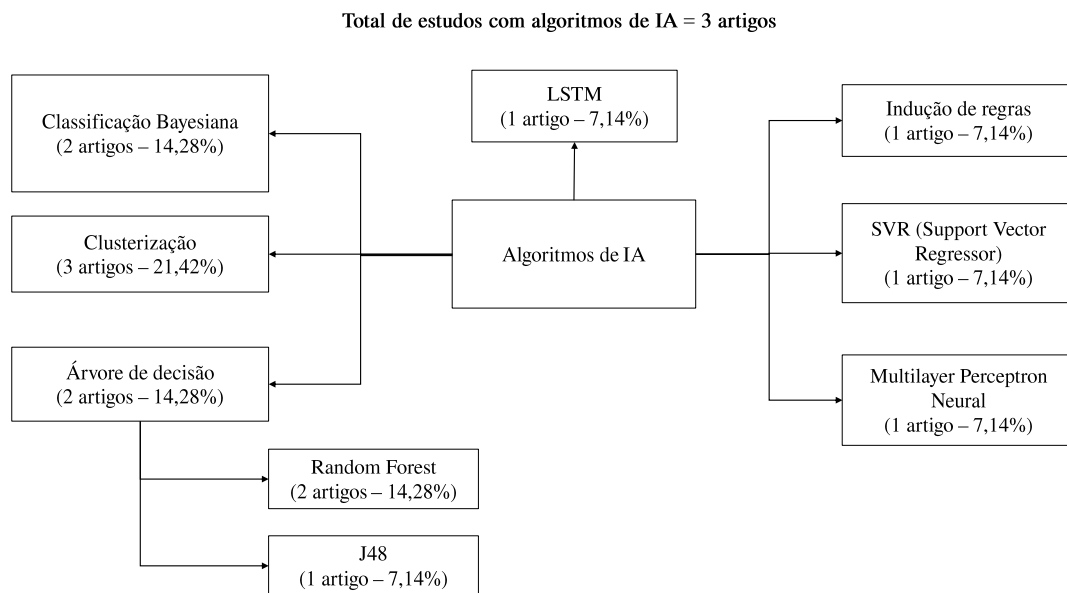


Figure 4. Algoritmos de inteligência artificial implementados nas publicações estudadas.

streaming. Também recomenda-se o uso moderado de redes sociais, apontadas como um fator de alto impacto no dreno de bateria. Essas aplicações, além de terem muitas funções de reprodução de mídia como vídeos, áudios e músicas, também contém uma alta carga de anúncios, acarretando em um custo energético elevado. Desligar os serviços de localização, Bluetooth e dados móveis quando estes não estão sendo utilizados ajuda a economizar bateria.

## 6 Conclusão

Este artigo apresentou uma Revisão Sistemática da Literatura destinada à identificação de métricas, abordagens e recomendações que permitam reduzir o consumo de energia em dispositivos Android. Todas as etapas que compõem uma revisão, como elaboração do protocolo, condução, extração e, por fim, a sumarização foram abordadas a fim de permitir a replicabilidade desta RSL e auxiliar pesquisadores em futuras pesquisas.

Para validar a primeira etapa de seleção de artigos, e assim evitar a introdução de vieses, foi adotado o critério Kappa de concordância entre pares através do cálculo do coeficiente Kappa de Cohen, o que resultou em um coeficiente Kappa de 0.815, que representa uma concordância quase perfeita entre os pesquisadores (Perez, 2020).

Notou-se que os usuários tendem a ser os maiores responsáveis pelo consumo de energia, uma vez que o tempo de uso, aplicativos preferidos e configurações personalizadas são fatores determinantes no dreno de bateria. Há consenso na literatura sobre a diversidade das opções do usuário de smartphones e seu impacto energético. Adquire especial relevância, nesse contexto, a exploração de trabalhos futuros com foco na análise do perfil do usuário, sobretudo pesquisas com bases largas de dados coletadas em tempo uso. A análise de publicações mais recentes permitiu perceber uma aplicação crescente de técnicas de inteligência artificial para adequar os aplicativos, de maneira automática, ao contexto

de uso visando economizar energia. Essas técnicas também vêm sendo utilizadas para mineração em bases de dados com métricas de energia, com o objetivo de buscar correlações entre fatores e o nível da bateria e possibilidades de otimização. O foco maior deste trabalho está de fato na identificação/sumarização dos resultados que visam não apenas responder as três questões de pesquisas elaboradas mas também permitir a descoberta de lacunas a serem investigadas no entorno do estudo sobre consumo de energia em dispositivos Android.

## Acknowledgements

Esta pesquisa foi realizada com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001 e, conforme previsto nos Arts. 21 e 22 do decreto nº 10.521/2020, foi parcialmente financiada pela Motorola Mobility Comércio de Produtos Eletrônicos Ltda e Flextronics da Amazônia Ltda, nos termos da Lei Federal nº 8.387/1991, através de convênio nº 004/2021, firmado com o ICOMP/UFAM.

## References

- Almasri, A. and Sameh, A. (2019). Rating google-play apps' energy consumption on android smartphones. In *2019 2nd IEEE Middle East and North Africa COMMUNICATIONS Conference (MENACOMM)*, pages 1–6.
- Barreto Neto, A. C. S., Farias, F., Mialaret, M. A. T., Cartaxo, B., Lima, P. A., and Maciel, P. R. M. (2020). Building energy consumption models based on smartphone user's usage patterns. *CoRR*, abs/2012.10246.
- Cañete, A., Horcas, J.-M., Ayala, I., and Fuentes, L. (2020). Energy efficient adaptation engines for android applications. *Information and Software Technology*, 118:106220.
- Dai, Z., Wang, W., and Wu, Y. (2020). Static energy consumption analysis for android applications. *IOP*

- Conference Series: Earth and Environmental Science*, 512:012011.
- Di Nucci, D., Palomba, F., Prota, A., Panichella, A., Zaidman, A., and Lucia, A. (2017). Petra: A software-based tool for estimating the energy profile of android applications.
- Dick, Zhuoqing Morley Mao, L. Y. (2011). Power tutor description. <https://developer.qualcomm.com/forum/qdn-forums/software/trepn-power-profiler/34230>. Accessed: 2021-11-24.
- Duan, L., Lawo, M., Rügge, I., and Yu, X. (2017). *Power Management of Smartphones Based on Device Usage Patterns*, pages 197–207.
- Elliot, J., Kor, a.-l., and Omotosho, O. (2017). Energy consumption in smartphones: An investigation of battery and energy consumption of media related applications on android smartphones.
- Google (2021). Battery stats description. <https://developer.android.com/topic/performance/power/setup-battery-historian>. Accessed: 2021-11-24.
- Guo, Y., Wang, C., and Chen, X. (2017). Understanding application-battery interactions on smartphones: A large-scale empirical study. *IEEE Access*, 5:13387–13400.
- Harihar, V. K. and Sukumaran, S. (2018). Behaviour comprehension and prediction using time series analysis of data for code offloading in mobile cloud computing. In *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pages 367–375.
- Kitchenham, B. A. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report.
- Malavolta, I., Grua, E. M., Lam, C.-Y., de Vries, R., Tan, F., Zielinski, E., Peters, M., and Kaandorp, L. (2020). Framework for the automatic execution of measurement-based experiments on android devices. *2020 35th IEEE/ACM International Conference on Automated Software Engineering Workshops (ASEW)*.
- Mehrotra, D., Srivastava, R., Nagpal, R., and Nagpal, D. (2021). Multiclass classification of mobile applications as per energy consumption. *Journal of King Saud University - Computer and Information Sciences*, 33(6):719–727.
- Pang, C., Hindle, A., Adams, B., and Hassan, A. E. (2016). What do programmers know about software energy consumption? *IEEE Software*, 33(03):83–89.
- Pereira, R., Matalonga, H., Couto, M., Castor, F., Cabral, B., Carvalho, P., Sousa, S., and Fernandes, J. (2021). Greenhub: a large-scale collaborative dataset to battery consumption analysis of android devices. *Empirical Software Engineering*, 26.
- Perez, J Diaz, B. T. (2020). Systematic literature reviews in software engineering—enhancement of the study selection process using cohen’s kappa statistic. *Journal of Systems and Software*, 168:110657.
- Qualcomm (2017). Trepn description. <https://developer.qualcomm.com/forum/qdn-forums/software/trepn-power-profiler/34230>. Accessed: 2021-11-24.
- Rua, R., Couto, M., and Saraiva, J. (2019). Greensource: A large-scale collection of android code, tests and energy metrics. pages 176–180.
- Statcounter (2021). Operating system market share worldwide. <https://gs.statcounter.com/os-market-share#monthly-200901-202111>. Acessado em 25/10/2021.
- UFSCar (2021). Start-lapes-ufscar. [http://lapes.dc.ufscar.br/tools/start\\_tool](http://lapes.dc.ufscar.br/tools/start_tool). Acessado em 25/10/2021.
- Wang, C., Guo, Y., Shen, P., and Chen, X. (2017). E-spector: Online energy inspection for android applications. pages 1–6.
- Zhang, X., Xiao, X., He, L., Ma, Y., Huang, Y., Liu, X., Xu, W., and Liu, C. (2019). Pifa: An intelligent phase identification and frequency adjustment framework for time-sensitive mobile computing. pages 54–64.