Building Blocks of RAG with Intel

RAG(Retrieve Augmented Generation)는 대규모 언어 모델(LLM)과 함께 조직이 데이터의 가치를 활용하는 방법을 정의하는 혁신적인 접근 방식입니다. RAG 애플리케이션을 최적화하여 상황에 맞는 실시간 대응을 가능하게 하는 동시에 구축을 단순화하고 확장을 가능하게 하는 Intel 하드웨어 및 소프트웨어 구성 요소를 살펴봅니다.

Authors: Eduardo Alvarez, Senior AI Solutions Engineer, Intel ▪ **Sancha Norris,** AI Product Marketing, Intel

애플리케이션에 맞게 GenAI 조정하기

ChatGPT의 등장은 인공지능의 판도를 바꾸었습니다. 기업들은 새로운 제품, 생산성 향상, 비용 효율적인 운영과 경쟁력을 갖추기 위해 이 새로운 기술의 활용을 서두르고 있습니다.

Grok-1 (매개변수 300억 개 이상) 및 GPT-4 (매개변수 수조 개 이상)와 같은 생성형 AI (GenAI) 모델은 방대한 양의 인터넷과 기타 텍스트 소스들로 부터 데이터를 학습합니다. 이러한 서드 파티(3rd Party) 의 대규모 언어 모델은 일반적이고 범용적인 사용 사례에 적합합니다. 그러나 대부분의 기업에서는 비즈니스와 더 연관성이 높은 결과를 얻기 위해 AI 모델을 추가로 학습시키거나 데이터로 증강시켜야 합니다. 다음은 다양한 산업 분야에서 생성형 AI를 어떻게 적용할 수 있는지에 대한 몇 가지 예시입니다.

소비재 및 소매업	헬스케어 및 제약	제조	미디어 & 엔터테인먼트	금융 서비스
▪ 가상 피팅룸 ▪ 배송 및 설치 ▪ 매장 내 제품 찾기 지원 ▪ 수요 예측 및 재고 계획 수립 ▪ 새로운 제품 디자인	▪ 바쁜 일선 직원 지원 ▪ 진료 노트의 기록 및 요약 ▪ 챗봇 및 의료 질문에 대한 답변 ▪ 진단 및 치료에 대한 정보를 제공하는 예측 분석	▪ 기술자를 위한 전문가 Copilot ▪ 기계와 대화형 상호작용 ▪ 규범적이고 사전 예방적인 현장 서비스 ▪ 자언어 문제 해결 ▪ 보증 현황 및 문서화 ▪ 프로세스 병목 현상 이해, 복구 전략 수립	▪ 지능형 검색, 맞춤형 콘텐츠 검색 ▪ 헤드라인 및 카피 개발 ▪ 콘텐츠 품질에 대한 실시간 피드백 ▪ 개인화된 재생 목록, 뉴스 요약, 추천 기능 ▪ 시청자 선택을 통한 인터랙티브 스토리텔링 ▪ 타겟에 맞춘 제안, 구독 플랜	▪ 트레이딩 신호 발견, 트레이더에게 취약한 포지션 알림 ▪ 증권 인수 의사 결정 가속화 ▪ 레거시 시스템 최적화 및 재구축 ▪ 뱅킹 및 보험 모델 리버스 엔지니어링 ▪ 잠재적 금융 범죄 및 사기에 대한 모니터링 ▪ 규정 준수를 위한 데이터 수집 자동화 ▪ 기업 공시에서 인사이트 추출

출처 : MIT Technology Review Insights, "Retail in the Age of Generative AI", [9] "The Great Unlock: Large Language Models in Manufacturing", [10] "Generative AI is Everything Everywhere, All at Once", "Large Language Models in Media & Entertainment"[12] Databricks, 2023년 4 ~ 6월.

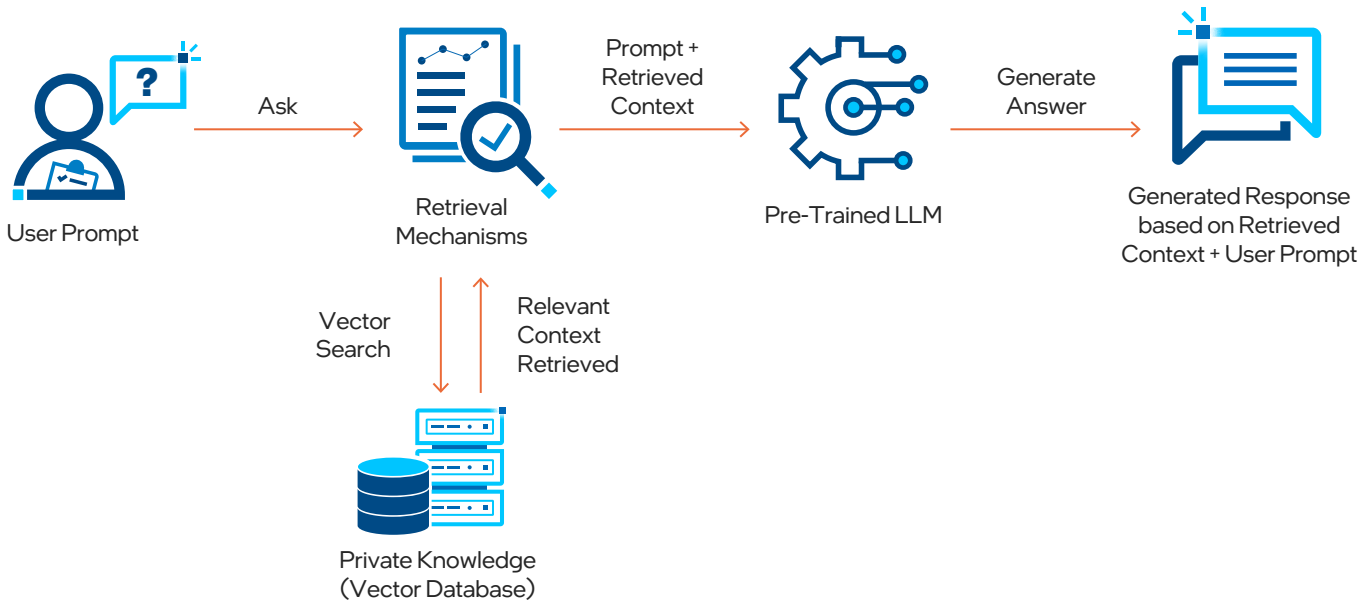
데이터를 사용하여 모델을 미세 조정할 수는 있지만, 모델을 재학습하려면 추가적인 시간과 리소스가 필요합니다. 또 다른 잘 알려진 기술인 RAG (검색 증강 생성)은 오픈소스 사전 학습 모델을 각 회사의 독점 데이터로 증강시키고 도메인에 특화된 LLM을 생성하여 비즈니스와 연관성이 높은 결과를 얻을 수 있는 기술입니다.

RAG를 사용하면 데이터를 타사의 대형 파운데이션 모델과 공유하지 않고도 안전하게 보호할 수 있습니다. 이 가이드에서는 RAG가 어떻게 인텔 최적화 된 다양한 플랫폼과 결합하여 GenAI 시스템에 놀라운 가치와 성능을 제공할 수 있는지에 대한 방법을 설명합니다.

RAG (검색 증강 생성) 란 무엇인가?

RAG 기술은 쿼리 종속적인 동적 데이터를 모델의 프롬프트 스트림에 추가합니다. 관련 데이터는 벡터 데이터베이스에 저장된 맞춤화된 지식 기반(Knowledge Base)에서 검색됩니다. 프롬프트와 검색된 컨텍스트는 모델의 결과물을 강화하여 보다 연관성이 높고 정확한 결과를 제공합니다. RAG를 사용하면 LLM을 통해 데이터를 활용하면서 데이터를 비공개로 유지하며 무결성을 유지할 수 있습니다. 왜냐하면, 모델을 관리하는 제3자에게 전송되지 않기 때문입니다.

RAG 워크플로우의 핵심 구성 요소는 사용자 쿼리 처리, 검색, 컨텍스트 통합 및 출력 생성의 네 가지 간단한 단계로 요약할 수 있습니다. 아래 다이어그램은 이 기본 흐름을 보여줍니다.



RAG의 유용성은 텍스트에만 국한되지 않으며, 비디오 검색과 대화형 문서 탐색에 혁신을 가져올 수도 있으며, 심지어 챗봇이 PDF 콘텐츠를 활용하여 답변을 얻을 수도 있습니다.

RAG 애플리케이션은 사용자 프롬프트에서 시작되는 일관된 데이터 프로세스 흐름으로 인해 종종 'RAG 파이프라인'이라고 불립니다. 프롬프트는 핵심 구성 요소인 검색 메커니즘을 통과하여 벡터 임베딩으로 변환하고, 미리 구성되어있는 벡터 데이터베이스(예: PDF, 로그, 트랜스크립트와 같은 디지털화된 문서 등)에서 관련된 콘텐츠를 찾기 위해 벡터 검색을 사용합니다.

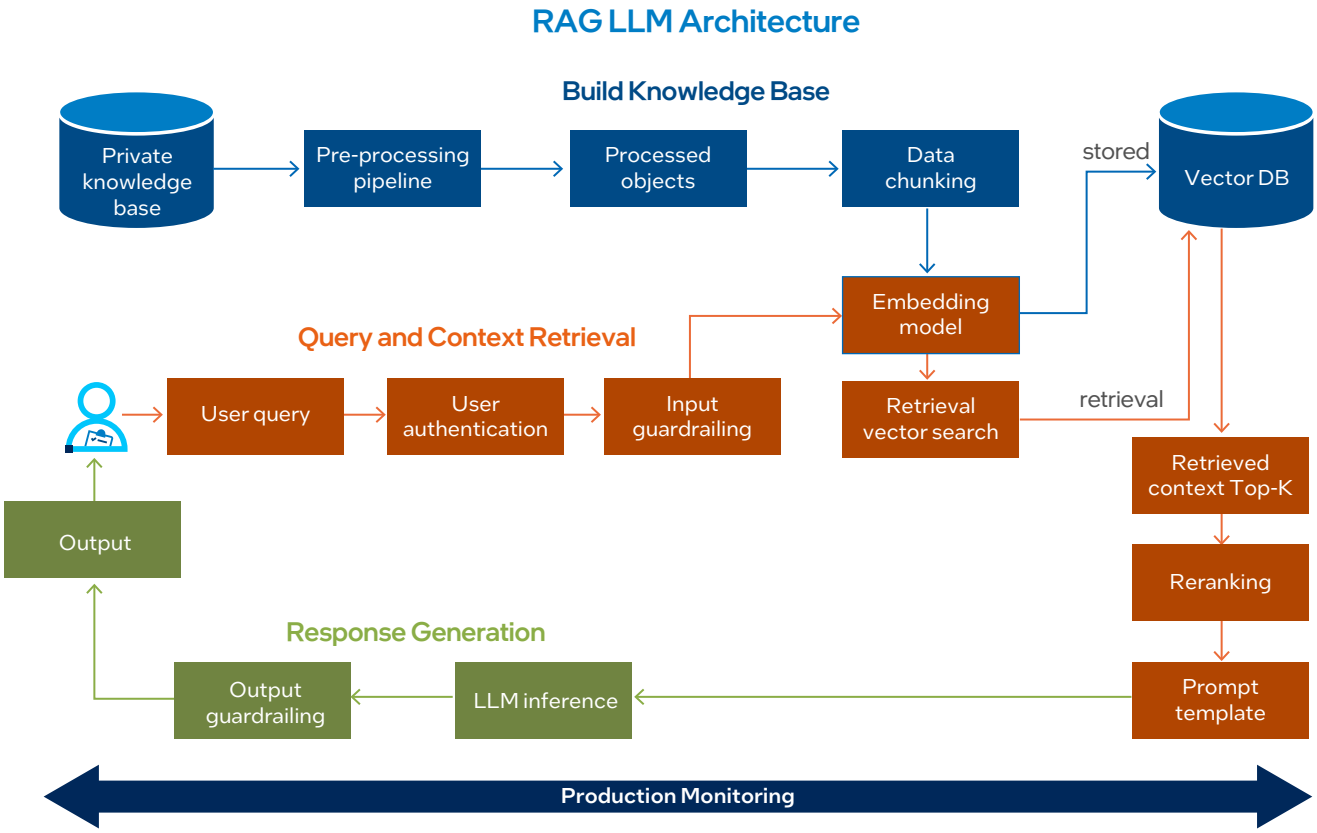
가장 연관성이 높은 데이터가 검색되고, 사용자의 프롬프트와 통합된 후 추론 서비스 및 최종 결과물의 생성을 위해 모델에 전달됩니다. 이러한 컨텍스트 통합은 사전 학습(pre-training) 중에 사용할 수 없었던 추가 정보를 모델에 제공하여 사용자의 작업 또는 관심 영역에 더 잘 맞출 수 있게 해줍니다.

왜냐하면, RAG는 모델을 재학습(retraining)하거나 미세 조정(fine-tuning) 할 필요가 없기 때문에 컨텍스트를 LLM에 제공하기 위해 데이터를 추가하는 것이 효율적인 방법이 될 수 있습니다.

다음 섹션에서는 RAG 솔루션 아키텍처와 스택에 대해 살펴봅니다.

표준 RAG 솔루션 아키텍처

다음 RAG 솔루션 아키텍처는 표준 RAG 구현의 빌딩 블록에 대한 개요를 제공합니다.
핵심 구성 요소에는 ❶ 지식 기반(Knowledge Base) 구축, ❷ 쿼리 및 컨텍스트 검색, ❸ 응답 생성, ❹ 애플리케이션 전반의 프로덕션 모니터링이 있습니다.



이러한 구성 요소를 자세히 살펴보겠습니다:

❶ 지식 베이스(Knowledge Base) 구축하기 :

- **데이터 수집 :** 트랜스크립트, PDF 및 디지털화 된 문서와 같은 텍스트 기반 소스에서 비공개 지식 베이스를 구축합니다.
- **데이터 처리 파이프라인 :** RAG 전용 파이프라인을 활용하여 텍스트를 추출하고, 처리할 콘텐츠의 형식을 지정하고, 데이터를 관리하기 쉬운 크기로 청크화 합니다.
- **벡터화 :** 임베딩 모델을 통해 청크를 처리하여 텍스트를 벡터로 변환합니다. 선택적으로 더 풍부한 컨텍스트를 위해 메타데이터를 포함할 수 있습니다.
- **벡터 데이터베이스 저장소 :** 벡터화된 데이터를 확장이 가능한 벡터 데이터베이스에 저장하여 효율적으로 검색할 수 있도록 합니다.

❷ 쿼리 및 컨텍스트 검색 :

- **쿼리 제출 :** 사용자 또는 하위 시스템은 채팅과 같은 인터페이스 또는 API 호출을 통해 쿼리를 제출하며, 보안 서비스에 의해 인증됩니다.
- **쿼리 처리 :** 보안 및 규정 준수를 위한 입력 안전 장치를 구현하고 쿼리 벡터화를 수행합니다.
- **벡터 검색 및 순위 재조정 :** 최초의 벡터 검색을 수행하여 관련 벡터를 검색한 다음, 보다 복잡한 모델을 사용하여 결과를 개선하기 위한 순위 재조정 프로세스를 수행합니다.

❸ 응답 생성 :

- **LLM 추론 및 응답 생성 :** 상위 컨텍스트를 사용자 쿼리와 결합하고, 사전 학습되거나 미세 조정된 LLM을 통해 처리하며, 품질과 안전을 위해 사후 처리합니다.
- **응답 전달 :** 인터페이스를 통해 최종 응답을 사용자 또는 하위 시스템에 반환하며, 일관성있고 맥락에 맞는 정확한 답변을 보장합니다.

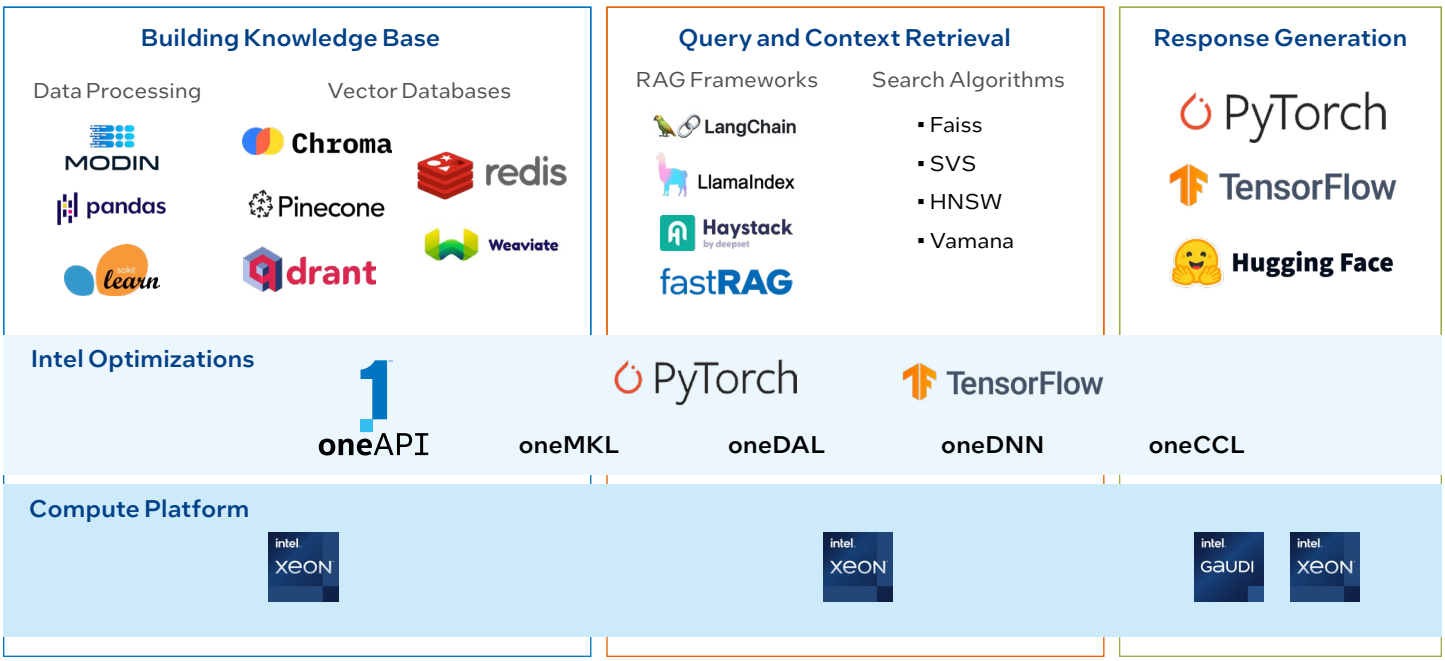
❹ 프로덕션 모니터링 :

- **검색 성능 :** 검색 프로세스의 지연 시간과 정확도를 모니터링하며, 감사 목적으로 기록을 보관합니다.
- **순위 재조정 효율성 :** 순위 재조정의 성능을 추적하여 문맥에 맞는 연관성과 속도를 보장합니다.
- **추론 서비스 품질 :** LLM 추론의 지연 시간과 품질을 관찰하고, 감사 및 개선을 위한 로그를 유지합니다.
- **가드레일 효율성 :** 입력 및 출력 처리를 위한 가드레일을 모니터링하며, 규정 준수 및 콘텐츠 안전을 보장합니다.

RAG 테크놀로지

RAG 애플리케이션 개발은 일반적으로 Haystack, LlamaIndex, LangChain, Intel Lab의 fastRAG와 같은 통합 RAG 프레임워크에서 시작됩니다. 이러한 프레임워크는 최적화를 제공하고 필수 AI 툴 체인을 통합하여 개발을 간소화합니다.

지식 기반 구축, 쿼리 및 컨텍스트 검색, 응답 생성이라는 익숙한 세 가지 구성 요소 측면에서 RAG 툴체인을 살펴봅시다. 종종 RAG 프레임워크는 전체 툴체인을 포괄하는 API를 제공합니다. 이러한 추상화를 사용할 것인지 독립적인 구성 요소를 활용할 것인지를 선택하는 것은 신중하게 고려해야 하는 엔지니어링 결정입니다.



인텔 최적화는 툴체인과 하드웨어 간의 격차를 해소하여 체인 전반의 성능을 향상하는 동시에 인텔® 제온® CPU 및 인텔® 가우디® 가속기에서의 호환성과 향상된 기능을 보장합니다. 이러한 최적화는 기본 프레임워크에 통합되거나 애드온 확장 기능으로 배포되어 광범위한 로우-레벨 프로그래밍의 필요성을 줄여줍니다. 이러한 추상화를 통해 개발자는 향상된 성능과 특정 사용 사례에 맞는 맞춤형 솔루션을 활용하여 RAG 애플리케이션을 효과적으로 구축하는데 집중할 수 있습니다.

이제 툴체인의 다양한 구성 요소를 더 자세히 살펴보겠습니다.

지식 베이스 구축 + 컨텍스트 검색:

- **통합 프레임워크 :** Haystack과 LangChain은 벡터 데이터베이스와 검색 알고리즘을 위한 높은 수준의 추상화를 제공하여 개발자가 Python 기반 환경 내에서 복잡한 프로세스를 관리할 수 있게 해주는 중요한 RAG 프레임워크입니다.

- **벡터 데이터베이스 기술 :** Pinecone, Redis, Chroma는 널리 사용되는 검색 알고리즘을 지원하는 주요 벡터 데이터베이스 솔루션입니다. 인텔 랩스(Intel Labs)의 SVS (Scalable Vector Search - 확장 가능한 벡터 검색)는 주요 벡터 데이터베이스와 통합될 것으로 예상되는 훌륭한 추가 기능입니다.

- **임베딩 및 모델 접근성 :** Hugging Face API를 통해 통합되는 임베딩 모델은 RAG 프레임워크에 원활하게 통합될 수 있어 고급 자연어 처리(NLP) 모델을 더 쉽게 포함할 수 있게 합니다.

응답 생성:

- **로우 레벨(Low-Level) 최적화 :** oneAPI 성능 라이브러리는 PyTorch, Tensor Flow, ONNX와 같은 인기 있는 AI 프레임워크를 최적화하므로 익숙한 오픈 소스 툴이 인텔® 하드웨어에 최적화되어 있다는 것을 알고 사용할 수 있습니다.

- **고급 추론 최적화 :** Intel® Extension for PyTorch와 같은 확장 기능은 고급 양자화 추론 기술을 추가하여 대규모 언어 모델의 성능을 향상시킵니다.

보시다시피 RAG에는 여러 개의 상호 연결된 구성 요소가 포함되며, 인텔 제온 CPU와 같은 단일 플랫폼에서 이를 관리하면 구성, 배포 및 유지 관리가 간소화됩니다. 대규모 LLM 또는 높은 처리량을 필요로 하는 LLM 추론의 경우, Gaudi 가속기를 활용하면 애플리케이션의 요구 사항을 충족시킬 수 있는 최적의 솔루션이 됩니다.

다음 섹션에서는 프로덕션 환경에서 RAG의 복잡성을 자세히 살펴보고 성공적으로 배포하는데 도움이 되는 다양한 고려 사항과 기술을 살펴보겠습니다.

프로덕션 환경에서 RAG 가속하기

RAG 파이프라인의 많은 구성 요소는 강도 높은 연산을 필요로 합니다. 하지만, 최종 사용자는 지연 시간이 짧은 응답을 요구합니다. 또한 RAG는 기밀 데이터에 사용되는 경우가 많기 때문에 전체 파이프라인을 안전하게 보호해야 합니다. 인텔의 기술은 RAG 파이프라인을 강화하여 컴퓨팅 플랫폼 전반에 걸쳐 성능을 확보하고 특정 도메인 및 산업에 맞춤화된 생성형 AI의 성능을 최대한 발휘할 수 있도록 지원합니다.

컴퓨팅 요구 사항

일반적으로 LLM 추론은 RAG 파이프라인에서 가장 계산 집약적인 단계이며, 특히 라이브 프로덕션 환경에서는 더욱 그렇습니다. 그러나, 초기 지식 베이스(Knowledge Base)를 만드는 작업 (데이터 처리 및 임베딩 생성)은 데이터의 복잡성과 볼륨에 따라 까다로운 작업일 수 있습니다. 일반 컴퓨팅에서의 인텔의 진보된 기술, AI 가속기 및 기밀 컴퓨팅의 발전은 데이터 프라이버시 및 보안을 보장하면서 전체 RAG 파이프라인의 컴퓨팅 문제를 해결하기 위한 필수 구성 요소를 제공합니다.
대부분의 소프트웨어 애플리케이션과 마찬가지로 RAG는 최종 사용자의 트랜잭션 요구사항을 충족하도록 맞춤화되고 확장 가능한 인프라의 이점을 제공합니다.

트랜잭션 수요가 증가함에 따라 개발자는 컴퓨팅 인프라의 부하 (벡터 데이터베이스 쿼리 및 추론 계산으로 인해 포화 상태가 됨) 로 인한 지연 시간 증가를 경험하게 됩니다. 이러한 이유로, 증가하는 수요를 신속하게 처리할 수 있도록 시스템을 확장하기 위해 즉시 사용 가능한 컴퓨팅 리소스에 액세스하는 것이 중요합니다. 마찬가지로 중요한 것은 임베딩 생성, 벡터 검색 및 추론과 같은 주요 구성 요소의 성능을 향상시키기 위해 중요한 최적화를 구현해야 하는 것입니다.

데이터 프라이버시 및 보안

- **안전한 AI 프로세싱 :** 인텔® 소프트웨어 가드 확장(인텔® SGX) 및 인텔® 트러스트 도메인 확장(인텔® TDX)은 AI 프로세싱 중 CPU 메모리에서 기밀 컴퓨팅 및 데이터 암호화를 통해 데이터 보안을 강화합니다. 이러한 기술은 민감한 정보를 처리하는 데 매우 중요하며, 파이프라인의 다양한 부분에 걸쳐 암호화된 데이터를 사용하여 안전한 RAG 애플리케이션을 생성하는 데 기여합니다. 이는 벡터 임베딩 생성, 검색 또는 추론 중에 민감한 데이터를 안전하게 처리해야 하는 RAG 애플리케이션에 필수적인 기능입니다.

- **적절한 가드레일링 구현 :** RAG 애플리케이션에서 가드레일링은 RAG 시스템 내에서 LLM의 동작을 관리하기 위한 조치를 구현하는 것을 포함합니다. 여기에는 모델의 응답을 모니터링하고, 가이드라인과 모범 사례를 준수하도록 돕고, 독설적인 위험, 부당한 편견, 개인정보 침해의 위험을 줄이기 위한 결과물 제어 등이 포함됩니다. RAG 애플리케이션의 가드레일링은 LLM이 시스템의 전반적인 목표와 요구 사항에 부합하도록 보장하면서 신뢰와 책임 있는 사용을 유지하는 데 도움이 됩니다.

오픈-소스 최적화

임베딩 최적화

- **양자화된 임베딩 모델 :** Intel Xeon 프로세서는 양자화된 임베딩 모델을 활용하여 문서에서 벡터 임베딩 생성을 최적화할 수 있습니다. 좋은 예로는 인텔® Neural Compressor로 양자화되고 Optimum-Intel과 호환되는 BAAI/BGE-small-en-v1.5 버전인 bge-small-en-v1.5-rag-int8-static이 있습니다.
대용량 텍스트 임베딩 벤치마크(MTEB)에서 양자화된 모델을 사용하여 검색 및 성능 작업의 순위를 재지정한 결과, 처리량을 향상시키면서 MTEB 성능 지표에서 부동 소수점(FP32)과 양자화된 INT8 버전 간의 차이가 2% 미만인 것으로 나타났습니다 (각주 1, 3 참조).

최근 Hugging Face와의 연구에서 초당 문서 처리량 측면에서 최대 인코딩 성능에 대한 처리량을 평가한 결과, 전반적으로 모든 모델 크기에서 양자화된 모델은 다양한 배치 크기에서 기준인 bfloat16(BF16) 모델에 비해 최대 4배까지 처리량이 향상된 것으로 나타났습니다. 자세한 내용은 여기에서 확인하세요: https://huggingface.co/blog/ intel-fast-embedding

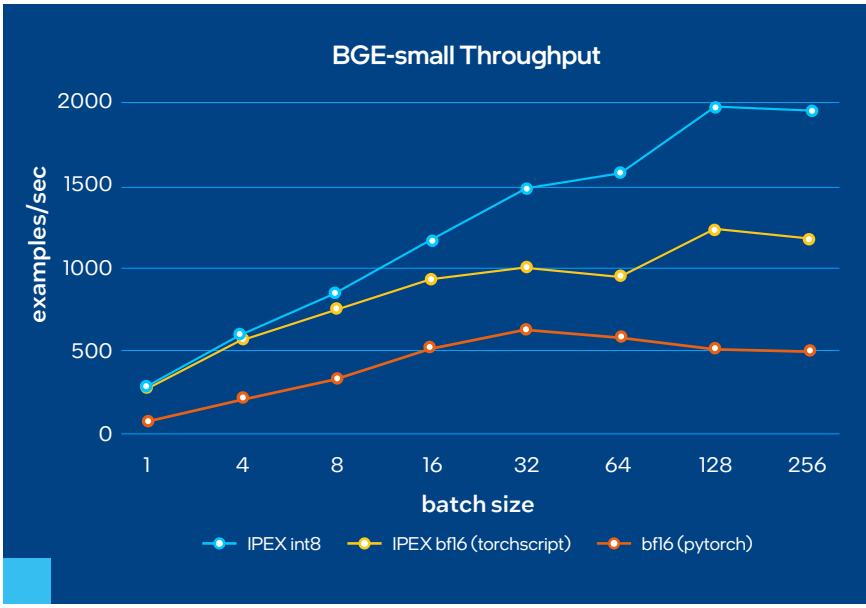


그림 1 : BGE small 의 처리량
출처 : https://huggingface.co/blog/intel-fast-embedding

벡터 검색 최적화

- **CPU에 최적화된 워크로드 :** 벡터 검색 작업은 인텔 제온 프로세서에서 고도로 최적화되어 있습니다. 3세대 이상의 프로세서에서 인텔® 고급 벡터 확장 512(Intel® AVX-512)가 도입되면서 특히 그렇습니다.
Intel AVX-512는 FMA(Fused Multiple-Add) 명령을 활용하여 단일 작업으로 곱셈과 덧셈을 결합하여 벡터 검색의 기본 연산인 내부 곱셈 계산을 향상시킵니다. 이 기능은 계산에 필요한 명령어 수를 줄임으로써 처리량과 성능을 크게 향상시킵니다.

- **SVS (Scalable Vector Search) :** SVS(확장 가능한 벡터 검색) 기술은 빠른 벡터 검색 기능을 제공하여 검색 시간을 최적화하고 전반적인 시스템 성능을 향상시킵니다. SVS는 정확도를 유지하면서 메모리 대역폭 요구 사항을 최소화하는 로컬 적응형 벡터 양자화(LVQ)를 사용하여 그래프 기반 유사성 검색을 최적화합니다. 그 결과 아래 그림에서 볼 수 있듯이 거리 계산 지연 시간이 크게 줄어들고 처리량과 메모리 요구 사항에서 더 높은 성능을 얻을 수 있습니다.

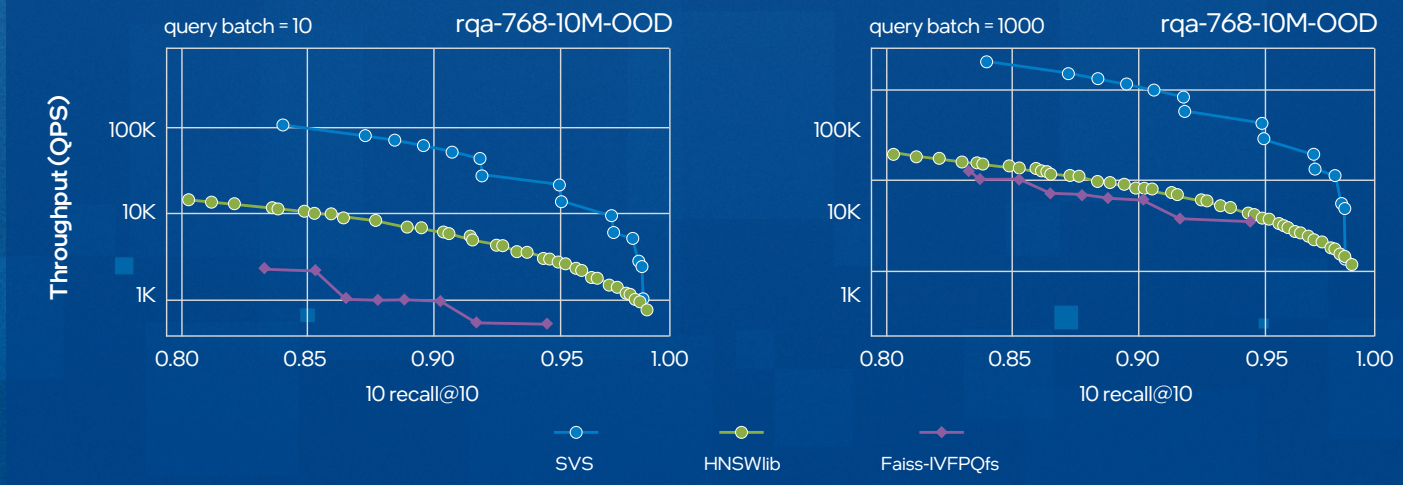


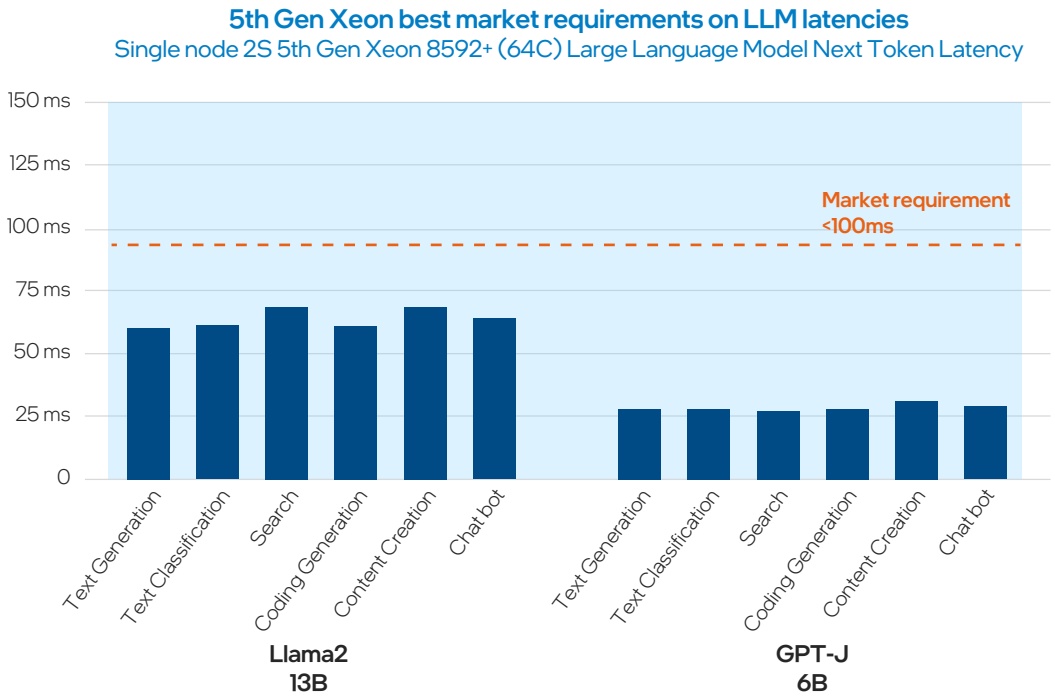
그림 2. 잘 채택된 다른 구현인 HNSWlib 및 FAISS와 비교한 SVS의 초당 쿼리 수(처리량) 성능. 그림은 rqa-768-10M-OOD 데이터 세트에 대한 QPS 대 리콜 곡선(배포 외 쿼리가 있는 조밀 통과 리트리버 모델 RocketQA[QDLL21]로 생성된 10M 768차원 임베딩)을 보여줍니다. (각주 2, 3)

출처 : https://intellabs.github.io/ScalableVectorSearch/benchs/static/latest.html

추론 최적화

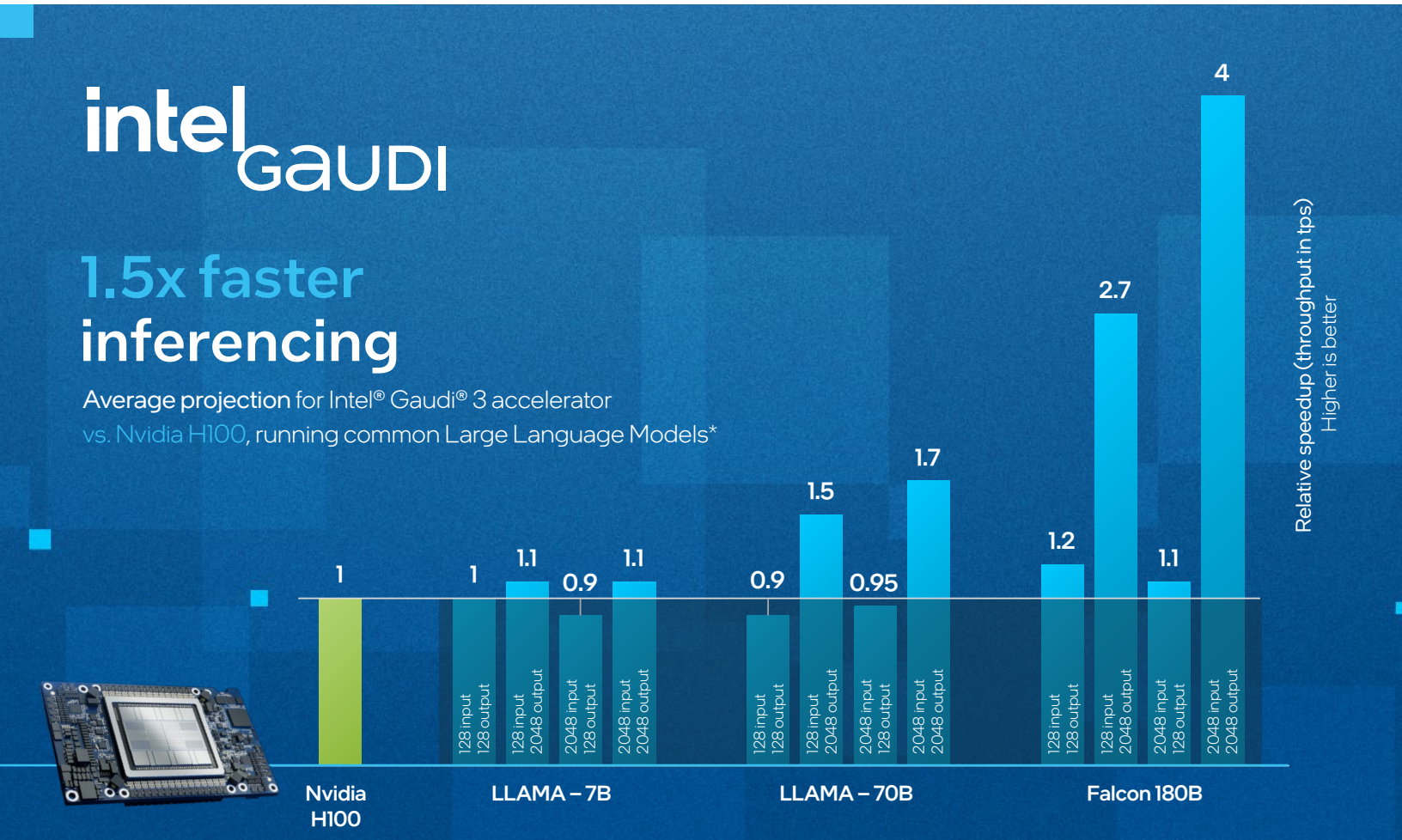
RAG에는 주로 추론 연산이 포함되며, 인텔 제온 프로세서는 고급 모델 압축 기술을 통해 이를 지원합니다. 이러한 기술을 통해 성능 저하 없이 낮은 정밀도(BF16 및 INT8)로도 연산을 수행할 수 있습니다. 더 큰 모델과 높은 처리량을 요구하는 경우, 인텔 Gaudi 가속기는 뛰어난 가격 대비 성능 이점을 제공하며 RAG 추론을 위해 CPU 및 기타 가속기를 대체할 수 있습니다. 이 섹션에서는 추론에 특화된 다양한 최적화와 기회에 대해 설명합니다.

- **인텔 AMX (Intel Advanced Matrix Extensions) :** 4세대 및 5세대 인텔 제온 스케일러블 프로세서는 인텔 AMX를 통합하여 보다 효율적인 매트릭스 작업과 향상된 메모리 관리가 가능합니다.
- **오픈 소스 최첨단 추론 최적화 도구 :** 인텔은 PyTorch, TensorFlow, Hugging Face, DeepSpeed 등과 같은 인기 있는 딥 러닝 프레임워크에 기여하고 확장합니다. RAG 워크플로우가 흥미로운 점은 양자화와 같은 모델 압축 기술을 구현하여 LLM을 최적화할 수 있다는 점입니다.
  Intel® Extension for PyTorch는 현재 SmoothQuant, 가중치 전용 양자화, 혼합 정밀도(FP32/BF16) 와 같은 다양한 최신(state-of-the-art) LLM 양자화 레시피를 제공합니다. 아래 그림은 4세대 인텔 제온 플랫폼 싱글 소켓에서 실행되는 INT8으로 양자화된 라마 2 모델의 추론 지연 시간의 성능을 보여줍니다.



See backup configuration for workload and configurations. Results may vary.

그림 3. 5세대 인텔® 제온® 스케일러블 프로세서의 라마 2 13B 및 GPT-J 6B 성능[3]



*출처 : https://nvidia.github.io/TensorRT-LLM/performance.html#h100-gpus-fp8, 에 기준한 NVidia H100 비교, 2024년 3월 28일.
성능수치는 LLAMA2-7B, LLAMA2-70B 및 Falcon 180B 프로젝션의 경우 GPU 당 Intel® Gaudi® 3 와의 성능 비교치입니다. 결과는 다를 수 있습니다.

그림 4. 인텔 가우디 3의 LLM 추론 성능

추론 복잡성과 인텔 가우디

RAG의 장점 중 하나는 LLM의 '지식'에 덜 의존하고 '언어 모델링 기능'에 더 많이 의존하기 때문에 훨씬 적은 수의 매개변수로 모델을 사용할 수 있다는 점입니다. 많은 경우, RAG를 사용하는 7억 개의 작은 매개변수 모델이 RAG 모델의 지식 베이스(Knowledge Base)와 관련된 도메인 특화된 작업에서 수백억 개의 매개변수를 사용하는 대형 모델을 능가할 수 있습니다.

고도로 전문화된 작업에는 때로는 더 큰 모델이 필요할 수 있으며, 따라서 인텔 가우디 프로세서와 같은 전문화된 가속기가 필요할 수 있습니다. 최고 처리량 또는 최저 지연 시간이 필요한 RAG 애플리케이션의 경우, 인텔 가우디 3 프로세서와 같이 최고의 성능을 발휘할 수 있는 AI 가속기에서 LLM 추론을 실행합니다.

자세한 내용은 Intel Gaudi RAG 리소스를 참조하십시오

- Intel Labs Cognitive AI 팀의 Intel® Vision 2024의 Multi-Modal RAG 데모
- Intel Gaudi®2 가속기를 사용하여 최적화된 애플리케이션을 배포하는 방법으로 Hugging Face 도구를 사용하는 확장 가능한 RAG(Retrieve Augmented Generation) 애플리케이션

기업에서 RAG 의 기회

리테일

소매업체는 다양하고 변화하는 고객의 선호도에 맞는 상품을 추천해야 하는 과제에 직면해 있습니다. 기존의 추천 시스템은 최신 트렌드나 개별 고객의 피드백을 효과적으로 반영하지 못해 관련성이 떨어지는 추천을 할 수 있습니다.

RAG 기반 추천 시스템을 구현하면 소매업체는 최신 트렌드와 개별 고객 피드백을 개인화된 제품 추천에 동적으로 통합할 수 있습니다. 이 시스템은 관련성 있고 시의적절한 개인화된 상품 추천을 제공함으로써 쇼핑 경험을 풍부하게 하여 매출과 고객 충성도를 높일 수 있습니다.

자세히 알아보기 ›

제조

제조업에서는 장비 고장으로 인한 예기치 않은 다운타임은 상당한 비용 요인이 됩니다. 기존의 예측 유지보수 모델은 특히 과거 고장 데이터가 제한적이거나 존재하지 않을 수 있는 복잡한 기계에서 고장이 발생하기 전의 미묘한 이상 징후를 놓칠 수 있습니다.

예측 유지보수를 위한 RAG 기반 이상 징후 감지 시스템은 방대한 양의 운영 데이터를 실시간으로 분석하여 장비 성능에 대한 광범위한 지식 기반과 비교함으로써 잠재적인 고장이 발생하기 전에 이를 식별할 수 있습니다. 이 접근 방식은 다운타임과 유지보수 비용을 최소화하는 동시에 장비 수명을 연장합니다.

자세히 알아보기 ›

금융 서비스

끊임없이 변화하는 방대한 양의 금융 데이터와 규정으로 인해 개인화된 금융 조언을 제공하는 것은 쉽지 않습니다. 고객은 기존 챗봇이 정확하게 제공할 수 없는 신속하고 관련성 높은 개인화된 금융 조언을 기대합니다.

RAG 모델은 최신 금융 데이터와 규정을 동적으로 가져와 개인화된 조언을 생성함으로써 금융 상담 챗봇을 향상시킵니다. 이 챗봇은 방대한 지식 베이스를 활용하여 고객에게 맞춤형 투자 전략, 실시간 시장 인사이트, 규제 관련 조언을 제공함으로써 고객 만족도와 참여도를 높일 수 있습니다.

자세히 알아보기 ›

다음 단계로 넘어가기

구현을 시작할 준비가 되면 인텔은 Intel® Tiber™ 개발자 클라우드의 하드웨어 액세스부터 구글 클라우드 플랫폼, 아마존 웹 서비스, 마이크로소프트 애저와 같은 주요 클라우드 제공업체의 유비쿼터스 컴퓨팅에 이르기까지 시작하는 데 도움이 되는 일련의 리소스를 제공합니다. 코드 샘플, 워크스루, 교육 등을 원하는 개발자는 Intel Developer Zone을 방문하세요.

Intel® Tiber™ 개발자 클라우드

최신 인텔® Xeon® 프로세서 및 GPU 컴퓨팅에서 인텔®에 최적화된 소프트웨어를 사용하여 AI 개발을 가속화합니다.

최신 인텔 ® Xeon® 프로세서, 인텔 ® Gaudi® Accelerator 및 기타 인텔 플랫폼에서 인텔 ® 최적화된 소프트웨어를 사용하여 AI 개발을 가속화합니다.

Intel 하드웨어에 액세스하고 Amazon Web Services, Google Cloud Platform 및 Microsoft Azure와 같은 클라우드 공급업체에서 RAG 애플리케이션 구축 시작하세요

aws

Google Cloud Platform

Azure

인텔® 하드웨어 및 소프트웨어 개발을 위한 공식 소스

인텔의 가장 인기 있는 개발 분야와 리소스를 살펴보세요

Intel GenAI Development Resources

intel dev zone

intel®

[1] Performance claims based on 4th gen Intel Xeon 8480+ with 2 sockets, 56 cores per socket. Pytorch model was evaluated with 56 cores on 1 CPU socket. IPEX/Optimum setups were evaluated with ipexrun, 1 CPU socket, and cores ranging from 22-56. TCMalloc was installed and defined as an environment variable in all runs. See www.intel.com/performanceindex for details. Results may vary.

[2] Performance claims based on a 2-socket 4th generation Intel® Xeon® Platinum 8480L CPU with 56 cores per socket, equipped with 512GB DDR4 memory per socket @4800MT/s speed, running Ubuntu 22.04.12 For the deep-96-1B, dataset we use a server with the same characteristics except that it is equipped with 1TB DDR4 memory per socket @4400MT/s speed. See www.intel.com/performanceindex for details. Results may vary.

[3] Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex. Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. No product or component can be absolutely secure. Your costs and results may vary. Intel technologies may require enabled hardware, software or service activation. © Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

0424/SN/MESH/PDF 358260-001US